

VŠB – Technická univerzita Ostrava
Fakulta strojní
Katedra automatizační techniky a řízení

Zpracování a prezentace experimentálních datových souborů

Processing and Presentation of Experimental Data Files

Student:	Bc. Petr Kalman
Vedoucí diplomové práce:	prof. Dr. RNDr. Lubomír Smutný
Konzultant diplomové práce:	Ing. David Fojtík, Ph.D.

Ostrava: 22. 5. 2009

Prohlášení studenta

Prohlašuji, že jsem celou diplomovou práci včetně příloh vypracoval samostatně pod vedením vedoucího diplomové práce a uvedl jsem všechny použité podklady a literaturu.

V Ostravě

.....
Petr Kalman

Prohlašuji, že

- byl jsem seznámen s tím, že na moji diplomovou práci se plně vztahuje zákon č. 121/2000 Sb. – autorský zákon, zejména §35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a §60 – školní dílo.
- беру на вѣдомі, же Высoká škola báňská – Technická univerzita Ostrava (dále jen VŠB-TUO) má právo nevýdělečně ke své vnitřní potřebě diplomovou práci užít (§35 odst. 3).
- souhlasím s tím, že jeden výtisk diplomové práce bude uložen v Ústřední knihovně VŠB-TUO k prezenčnímu nahlédnutí a jeden výtisk bude uložen u vedoucího diplomové práce. Souhlasím s tím, že údaje o diplomové práci, obsažené v Záznamu o závěrečné práci, umístěném v příloze mé diplomové práce, budou zveřejněny v informačním systému VŠB-TUO.
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu §12 odst. 4 autorského zákona.
- bylo sjednáno, že užít své dílo – diplomovou práci nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).
- беру на вѣдомі, же оdevzdáním své diplomové práce souhlasím se zveřejněním této práce podle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, bez ohledu na výsledek její obhajoby.

V Ostravě

.....
Petr Kalman

Adresa trvalého pobytu

Petr Kalman
Patrice Lumumby 68
Ostrava-Jih
700 30

ANOTACE DIPLOMOVÉ PRÁCE

Kalman, P. *Zpracování a prezentace experimentálních datových souborů*

Ostrava: Katedra Automatizační techniky a řízení, Fakulta strojní

VŠB – Technická univerzita Ostrava, 2009, 70 s. Diplomová práce, vedoucí Smutný, L.

Diplomová práce se zabývá zpracováním a prezentací experimentálních datových souborů, které jsou získávány z programu WinGAp-CTRL. V první části práce je seznámení s předchozími programovými moduly, vyvinutými na katedře Automatizační techniky a řízení. Jsou rozebrány jejich vlastnosti a ověřena funkčnost.

Další část práce obsahuje návrhy programového řešení pro novou aplikaci, shrnutí možností jednotlivých řešení a výběr jednoho z nich. Je zvolen doplněk pro Microsoft Excel na aplikační úrovni s použitím jazyka Visual Basic a platformy .NET. Následující část je zaměřena na tvorbu Ribbon prvku, který je součástí Office 2007 jako nové uživatelské rozhraní. Tvorba tohoto prvku je v práci popsána převážně pomocí XML dokumentu.

V závěru je uvedena samotná dokumentace nově vytvořeného doplňku a příklad ověření funkčnosti s reálnými experimentálními daty. Jsou popsány metody aproximace experimentálních datových souborů použité v doplňku a je naznačen další možný vývoj této aplikace.

ANNOTATION OF THE THESIS

Kalman, P. *Processing and Presentation of Experimental Data Files*

Ostrava: Department of Control Systems and Instrumentation, Faculty of Mechanical Engineering

VSB – Technical University of Ostrava, 2009, 70 p. The thesis, head Smutný, L.

The thesis deals with processing and presentation of experimental data files acquired from WinGAp-CTRL application. The first part of this work is introduction to previous versions of program modules developed at Department of Control Systems and Instrumentation. Their features and functionalities are analyzed and verified.

Next part of this work contains the proposals of program solution for a new application, summary of each solution's possibilities, from which one is selected. An application level add-in for Microsoft Excel is chosen there with use a Visual Basic programming language and .NET Framework. Next part is focused on creating the Ribbon control, which is a part of Office 2007 as a new user interface. Creating this control is described mainly by XML document.

At the end, there is written documentation for a new add-in and an example of functionality verification with real experimental data files. The methods of approximation data files applied in add-in are described there and further possible progress of this application is proposed.

Obsah

SEZNAM POUŽITÝCH ZKRATEK	7
1. ÚVOD	8
2. ANALÝZA MOŽNÝCH POŽADAVKŮ NA PROGRAMOVÝ MODUL.....	9
3. FUNKČNOST REALIZOVANÝCH PROGRAMOVÝCH MODULŮ	11
3.1 PROGRAMOVÝ MODUL APROX.....	11
3.2 PROGRAMOVÝ MODUL WINGAP.....	15
3.3 PROGRAMOVÝ MODUL JAP	19
3.4 SHRNUÍ VLASTNOSTÍ PROGRAMOVÝCH MODULŮ.....	22
4. NÁVRHY PROGRAMOVÉHO ŘEŠENÍ.....	24
4.1 ARCHITEKTURA .NET	24
4.2 DOPLŇK PRO MS EXCEL – VB.NET.....	27
4.3 APLIKACE WIN32 – VB.NET.....	28
4.4 SERVEROVÁ APLIKACE ASP.NET – VB.NET	29
4.5 SHRNUÍ MOŽNOSTÍ PROGRAMOVÝCH ŘEŠENÍ	30
5. VISUAL STUDIO 2008.....	31
5.1 VISUAL STUDIO TOOLS FOR OFFICE	31
5.2 ARCHITEKTURA VSTO 2008	35
6. RIBBON PRVEK.....	38
6.1 MOŽNOSTI PŘIZPŮSOBENÍ A OVLÁDACÍ PRVKY	39
6.2 XML SCHÉMA A XML FORMÁT	40
6.3 DYNAMICKÁ AKTUALIZACE.....	42
6.4 PROJEKTY PRO RIBBON PRVKY	43
6.5 TIPY PRO PRÁCI S RIBBON PRVKEM.....	44
7. DOKUMENTACE K DOPLŇKU VISUAL APROX.....	47
7.1 POPIS NOVÉHO DOPLŇKU	47
7.2 POŽADAVKY, VLASTNOSTI A MOŽNOSTI VYTVOŘENÉHO DOPLŇKU	52
8. APROXIMACE EXPERIMENTÁLNÍCH DAT	58
9. ZÁVĚR.....	60
POUŽITÁ LITERATURA.....	62
SEZNAM OBRÁZKŮ	65
SEZNAM TABULEK.....	66
PŘÍLOHY:	67

Seznam použitých zkratk

ADO.NET	–	ActiveX Data Object – Model pro práci s daty v .NET
API	–	Application Programming Interface – Rozhraní pro programování aplikací
ASP	–	Active Server Pages – Aktivní serverové stránky
ASP.NET	–	Active Server Pages .NET – Aktivní serverové stránky s nadstavbou .NET pro tvorbu webových aplikací a služeb
BCL	–	Base Class Library – Knihovna tříd v .NET
CLR	–	Common Language Runtime – Společný jazykový běhový modul v .NET
COM	–	Component Object Model – Objektový model
GUI	–	Graphic User Interface – Grafické uživatelské rozhraní
HTML	–	HyperText Markup Language – Jazyk používaný pro tvorbu hypertextových dokumentů
IL	–	Intermediate Language – Mezikód .NET Frameworku
MAF	–	Managed Add-in Framework – Framework, který je součástí běhového prostředí <i>Visual Studio Tools for Office</i> , zajišťuje mapování mezi .NET a COM objekty
MS-DOS	–	Microsoft-Disc Operating System – Diskový operační systém společnosti Microsoft
PDA	–	Personal Digital Assistant – Osobní digitální asistent
PIA	–	Primary Interop Assemblies – Mechanismus pro psaní <i>managed</i> (řízeného) kódu pro Office 2007
VB.NET	–	Visual Basic .NET – Programovací jazyk Visual Basic s nadstavbou .NET pro tvorbu jakýchkoliv aplikací
VBA	–	Visual Basic for Application – Programovací jazyk Visual Basic určený pro aplikace (př. Microsoft Excel, Microsoft Word)
VPC	–	Virtual Personal Computer – Virtuální osobní počítač
VSTO	–	Visual Studio Tools for Office – Nástroj pro tvorbu vlastních Office řešení
XML	–	Extensible Markup Language – Rozšiřitelný značkovací jazyk

1. Úvod

Práce se zabývá zpracováním a prezentací experimentálně získaných dat. V první části je obecný přehled všech možných požadavků na programový modul jako takový. Nastíněny jsou systémové požadavky programů, nároky na vstupy a výstupy, uživatelské prostředí a nakonec také na zpracování a prezentaci samotných dat.

V další části jsou popsány důležité vlastnosti, které by měl nový programový modul obsahovat. Poté je uveden konkrétní rozbor jednotlivých programových modulů, realizovaných na katedře Automatizační techniky a řízení, VŠB-TU Ostrava. V každé podkapitole jsou uvedeny systémové požadavky, vstupy a výstupy programu, uživatelské prostředí a hlavně možnosti zpracování a prezentace dat. Vše je pro přehlednost a porovnání doplněno obrázky (převážně uživatelského prostředí a grafického výstupu zpracovaných dat na obrazovku).

Následující kapitola poskytuje návrhy programového řešení, které se zabývají především .NET technologií a to jak v podobě doplňku pro aplikaci Microsoft Excel, tak pro samostatnou WIN32 aplikaci a také pro webové řešení. Na začátku kapitoly je proto popsána celá .NET architektura a jsou vysvětleny pojmy jako *Intermediate Language*, *.NET Framework*, *Common Language Runtime* a *Base Class Library*, neboť všechna navrhovaná řešení se o ni přímo opírají. V jednotlivých podkapitolách se pak nepřímo vyskytují výhody a nevýhody navržených řešení, postupy, jakými se daná řešení tvoří a požadavky, které jsou zapotřebí k nasazení vybraného řešení. V poslední podkapitole jsou zdůrazněny převážně výhody pro konkrétní použití (v laboratořích katedry Automatizační techniky a řízení) a je vybrána první navržená možnost, vytvořit doplněk pro Microsoft Excel na aplikační úrovni, s použitím programovacího jazykem *Visual Basic .NET*.

Z tohoto důvodu je v následující části krátce představeno nové *Visual Studio 2008*, jsou uvedeny novinky, které se v něm nacházejí a hlavně je v této části popsán nástroj *Visual Studio Tools for Office*, pomocí kterého lze vytvářet vlastní Office řešení s využitím platformy .NET. Je popsána tvorba a použití nového Ribbon prvku, který v předchozích verzích aplikačního balíku Microsoft Office nebyl zastoupen.

V další části této práce je samotná dokumentace vytvořeného doplňku a příklad ověření funkčnosti s reálnými experimentálními daty. Jsou popsány metody aproximace experimentálních datových souborů použité v doplňku a je naznačen další možný vývoj této aplikace.

2. Analýza možných požadavků na programový modul

V současné době existuje spousta programů běžících na různých platformách. Z tohoto a dalších důvodů je potřeba analyzovat požadavky, vlastnosti a možnosti programových modulů, které jsou dostupné. Už teď je však jasné, že obecně není možno určit optimální řešení programu. Pouze s přihlédnutím na konkrétní použití aplikace (zpracování a prezentace experimentálních datových souborů) lze zvolit vhodné programové řešení.

Hlavním rozdílem mezi aplikacemi je běhové prostředí. Pokud existuje program pracující pod operačním systémem Windows, je potřeba jej nainstalovat. Jestliže se jedná o řešení realizované prostřednictvím internetu a webového prohlížeče, je zde nutnost doinstalovat určité doplňky. Proto jsou tyto dvě řešení srovnatelné. V následující části jsou rozebrány všechny možné požadavky na program.

Systémové požadavky:

- Licence (popř. omezení)
- Hardware
- Software
- Operační systém, pod kterým aplikace pracuje
- Instalace programového modulu, nebo jeho součástí
- Připojení k síti internet

Požadavky na uživatelské prostředí:

- Vzhled aplikace a rozvržení prvků pracovní plochy
- Jednoduchost ovládání
- Grafické výstupy na obrazovku
- Práce s grafy
- Nápověda

Požadavky na vstupy a výstupy:

- Způsob načítání a ukládání dat
- Práce se soubory a grafy
- Podpora formátu souborů používaných v předchozích programových modulech
- Vlastní a univerzální formát souboru
- Maximální možný počet dat určených ke zpracování
- Možnost editace grafů a jejich export do textových editorů (např. Microsoft Word)
- Tiskový výstup

Požadavky na zpracování dat:

- Prokládání zadanych či načtených dat aproximačními křivkami
- Výpočet odchylek, korelací, střední odchylky apod.
- Výpočet přechodových charakteristik podle zvolených parametrů

Požadavky na prezentaci dat:

- Graf zadanych nebo načtených hodnot
- Graf aproximace
- Graf odchylek od aproximace
- Vykreslení více typů aproximace přes sebe do jednoho grafu
- Způsob a možnosti zobrazení grafů

3. Funkčnost realizovaných programových modulů

Na katedře Automatizační techniky a řízení, VŠB-TU Ostrava byla vytvořena řada programových modulů pro zpracování a prezentaci experimentálních datových souborů. Některé z nich jsou však zastaralé, pracují pouze pod starými operačními systémy, nebo jen nevyhovují představám studentů a potřebám výuky. Optimální programový modul určený pro zpracování a prezentaci experimentálních datových souborů by měl zajišťovat efektivní a jednoduchou práci s naměřenými daty a podporovat prezentaci těchto dat v podobě přehledných grafů. Měl by také umožňovat výpočet a vykreslení přechodových charakteristik podle zadaných parametrů, poskytovat možnost aproximace zadaných či naměřených dat a vykreslovat grafy aproximace a odchylek.

V ideálním případě by měla aplikace povolovat kreslení grafů „přes sebe“ nebo vykreslení více průběhů do jednoho grafu, aby bylo zřejmé, který průběh nebo aproximace je v daném případě nejlepší. Užitečné a praktické je také umožnit úpravy grafů a export do textového editoru, např. Microsoft Word. Mezi základní typy programového řešení patří WIN32 aplikace pro operační systémy Windows a aplikace typu klient-server, ke kterým se přistupuje přes internet.

3.1 Programový modul *APROX*

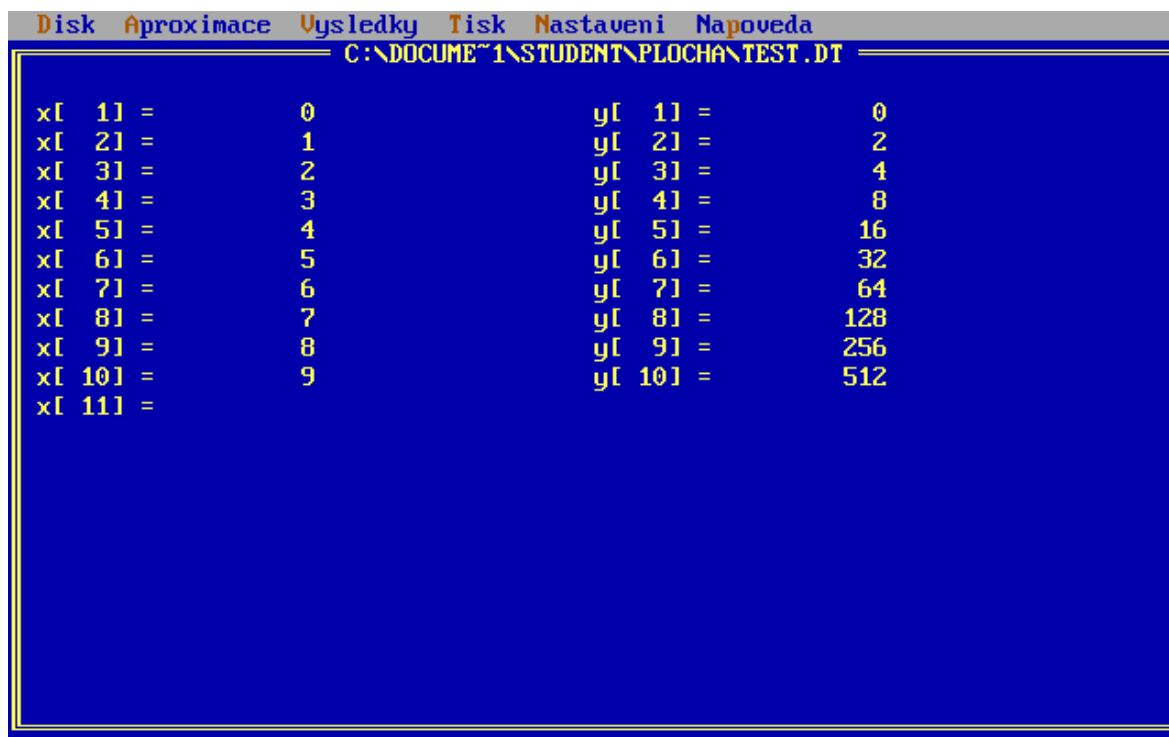
Tento modul slouží pro zpracování a prezentaci zadaných dat. Jeho původní název byl ZTK. Umožňuje aproximaci dat a také výpočet absolutní odchylky. Data je možné zadávat ručně, nebo načíst z vlastního formátu souboru s příponou **.dt*. Testování tohoto modulu probíhalo na počítači s operačním systémem Windows Vista, proto jsou všechny níže uvedené obrázky pořízeny *hard copy* obrazovky z virtuálního počítače (VPC), kde byl nainstalován OS Windows XP, pod kterým bylo možné aplikaci spustit z příkazové řádky.

Systémové požadavky:

Program je volně dostupný a šířitelný. Nemá žádné licenční omezení. Nároky na hardware a software jsou minimální, poněvadž se jedná o zastaralý programový modul. K provozu stačí jen běžný osobní počítač s grafickou kartou. Jelikož byla aplikace vytvořena v programovacím jazyce Turbo Pascal, je zřejmé, že je stavěna pro OS MS-DOS a proto není potřeba ji instalovat. Při práci nemusí být uživatel připojen k síti internet.

Uživatelské prostředí:

Aplikace je velmi přehledná a jednoduchá na ovládání díky dobře rozvrženému menu. Poskytuje také horké klávesy pro rychlejší ovládání a stručnou nápovědu, která snad není ani potřeba. Na druhou stranu je ovládání programu možné pouze pomocí klávesnice, což je pro systémy MS-DOS běžné, avšak pro uživatele méně komfortní.



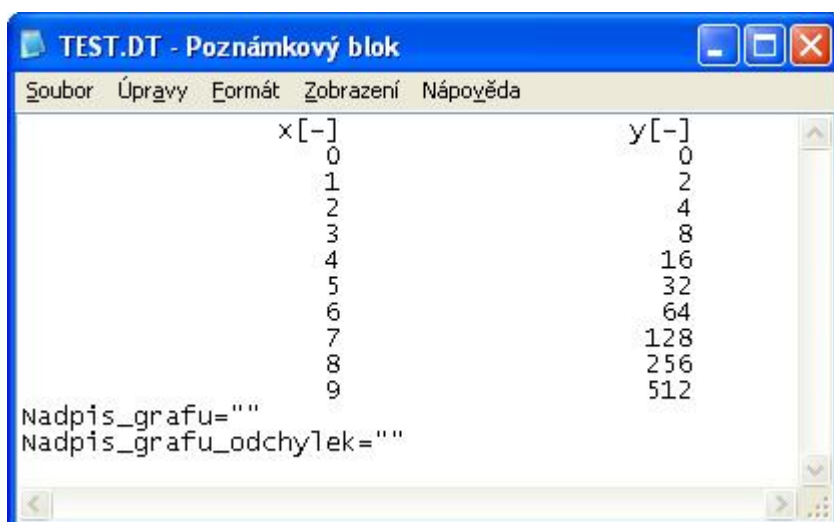
The screenshot shows the APROX program interface. At the top is a menu bar with options: Disk, Aproximace, Vysledky, Tisk, Nastaveni, and Napoveda. Below the menu bar is a title bar indicating the file path: C:\DOCUMENT1\STUDENT\PLOCHA\TEST.DT. The main area displays a table of data with two columns of values.

C:\DOCUMENT1\STUDENT\PLOCHA\TEST.DT			
x[1] =	0	y[1] =	0
x[2] =	1	y[2] =	2
x[3] =	2	y[3] =	4
x[4] =	3	y[4] =	8
x[5] =	4	y[5] =	16
x[6] =	5	y[6] =	32
x[7] =	6	y[7] =	64
x[8] =	7	y[8] =	128
x[9] =	8	y[9] =	256
x[10] =	9	y[10] =	512
x[11] =			

Obr. 1 Pracovní prostředí programu APROX se zadanými daty

Vstupy a výstupy:

Hodnoty lze zadávat do dvou sloupců (viz obr. 1). První z nich je nezávisle proměnná a druhý závisle proměnná. Program tedy pracuje s jednou nezávisle proměnnou a pouze s jednou závisle proměnnou. Počet zadaných hodnot v jednom sloupci je omezen na maximálně 450 hodnot [BRODA, R. 2005]. Program má pouze svůj vlastní formát souboru označený *.dt a nepodporuje více formátů. Data uložená v takovémto formátu souboru jsou v textové podobě a proto je lze otevřít v poznámkovém bloku, kde jsou vidět také informace o názvu grafu aproximace a grafu odchylek, uvedených v uvozovkách (viz obr. 2). Je možné popisovat také názvy os spolu s jednotkami.



x [-]	y [-]
0	0
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	256
9	512

Nadpis_grafu=""
Nadpis_grafu_odchylek=""

Obr. 2 Formát výstupu dat z programu APROX

Grafy lze nejen vykreslovat, ale i tisknout (na 9 nebo 24 jehličkových tiskárnách značky Epson a jim kompatibilních) a měnit měřítko tisku. Stejně tak lze tisknout aproximovaná data spolu s vypočtenými odchylkami a aproximačními koeficienty. Podpora tisku je však špatná, bez možnosti využití jiných či moderních tiskáren s vyšším rozlišením a nevýhodou je chybějící diakritika [LIPUS, M. 1999].

Zpracování dat:

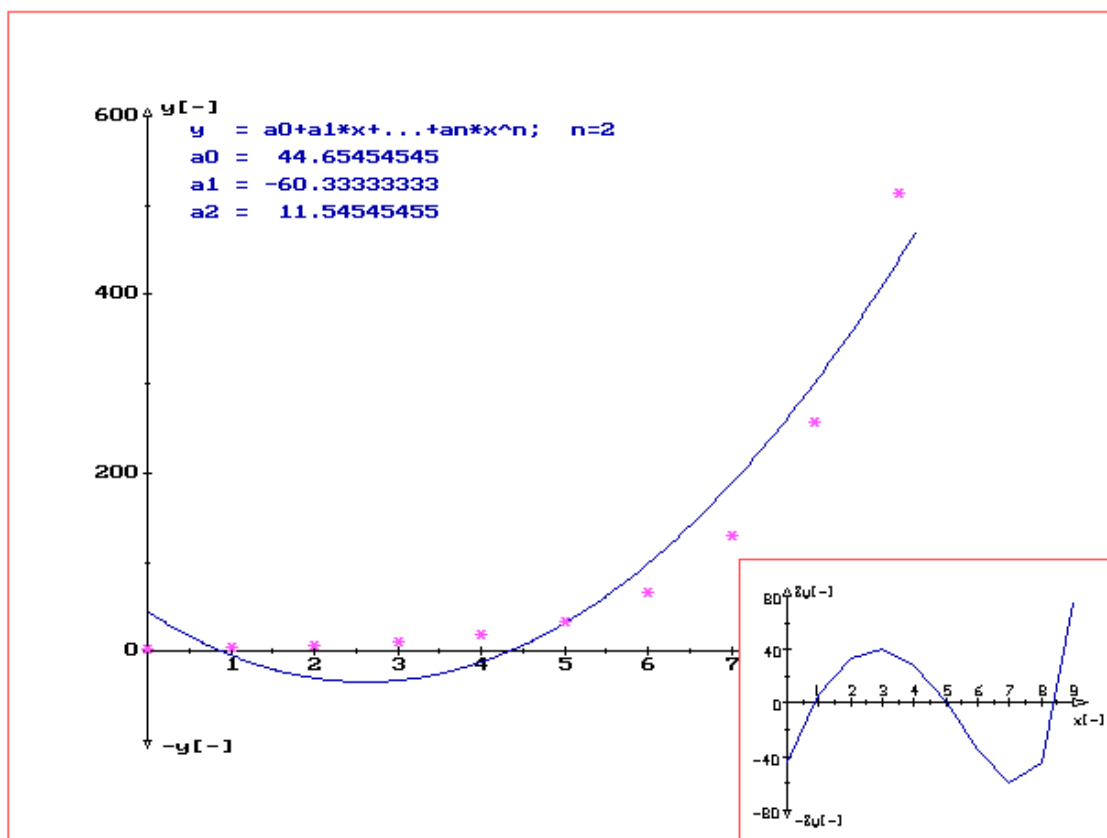
Aplikace umožňuje následující typy aproximace zadaných dat:

- přímka – jeden úsek (s volbou počátku a konce)
- přímka – dva úseky (s volbou počátků a konců)
- polynom $y = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$ pro $n \in \langle 1, 8 \rangle$
- polynom $y = a_0 + \frac{a_1}{x} + \frac{a_2}{x^2} + \dots + \frac{a_n}{x^n}$ pro $n \in \langle 1, 8 \rangle$
- funkce $y = a.e^{b.x}$
- funkce $y = a \cdot \left(1 - \frac{1}{e^{\frac{x}{b}}} \right)$
- funkce $y = a.x^b$
- funkce $y = \frac{x}{a + b.x}$

V případě aproximace pomocí přímky je možno zadat její počáteční a koncový bod. Pro aproximace pomocí polynomu má uživatel možnost zvolit stupeň polynomu ($n \in \langle 1, 8 \rangle$).

Zobrazení dat:

Program poskytuje různé typy zobrazení dat. Je možné zobrazit graf aproximace spolu s hodnotami (vykreslenými jako body), ze kterých byla aproximace provedena. Další možností je pak zobrazení grafu odchylek aproximace od výchozích dat. Velmi praktickým a zároveň dalším typem zobrazení je přiřazení jednoho grafu k druhému. Tzn., že lze vykreslit graf aproximace spolu s grafem odchylek aproximace současně a také naopak. Jeden z grafů je přitom majoritní (zobrazen větší) a druhý minoritní (zobrazen menší, v pravém dolním rohu obrazovky). Toto zobrazení je vidět na obr. 3, avšak z důvodu tisku této práce a úspory inkoustu jsou invertovány barvy, protože zobrazení grafu je velice nevhodné (s černým podkladem).



Obr. 3 Zobrazení zadaných dat (body) a jejich aproximace spolu s křivkou odchylek

Mimo jiné lze na obrazovku vypsát také tabulku hodnot obsahující zadané či načtené údaje o závislé a nezávislé ose a odchylky. Dále je možno vypsát parametry aproximace, jako jsou koeficienty aproximační funkce, index korelace, přiléhavost, střední odchylka a maximální odchylka.

V záložce nastavení je umožněno měnit názvy grafů, popisky os a jejich jednotky, manuálně nastavovat rozsah os, nebo ponechat automatické nastavení. Vyskytuje se zde i

možnost pro povolení či zákaz zobrazení aproximační funkce s jejími vypočtenými koeficienty. Toto uživatelsky definované nastavení programu je možno uložit do nebo načíst z konfiguračního souboru.

3.2 Programový modul WinGAp

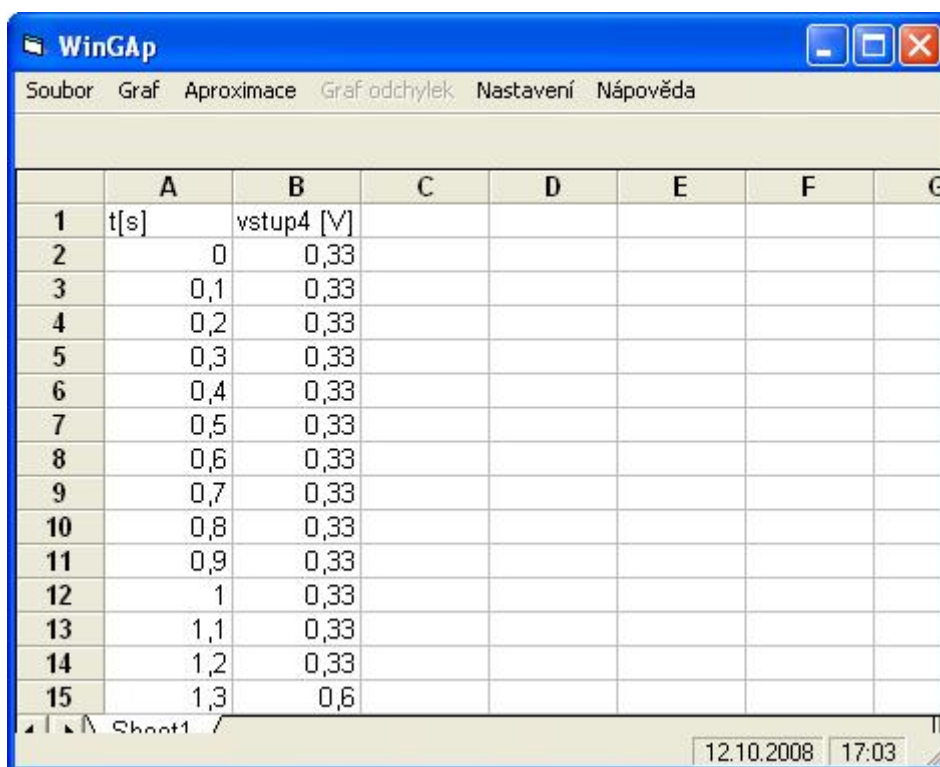
Tento modul slouží pro zpracování a prezentaci zadaných či experimentálně získaných dat. Umožňuje aproximaci naměřených dat a výpočet absolutní odchylky. Aplikace byla navržena kvůli novějším, graficky orientovaným OS jako jsou Windows. Využívá tedy vlastností OS Windows a je s nimi plně kompatibilní. Modul je vytvořen v jazyce Visual Basic 5.0, což znamená, že využívá grafické uživatelské rozhraní (GUI).

Systémové požadavky:

Program je volně dostupný a šířitelný. Nemá žádné licenční omezení. Nároky na hardware jsou určeny pouze OS, pod kterým aplikace pracuje, poněvadž využívá jeho základní funkce. Program je vytvořen v programovacím jazyce Visual Basic 5.0 a vnitřně používá nestandardních komponent, proto je potřeba jej instalovat. Tyto nestandardní komponenty jsou součástí instalačního balíčku. Důležitým upozorněním je, že pro správnou funkci tohoto programu musí být v národním prostředí Windows nastaven oddělovač desetinných míst čísla jako čárka a nesmí být nastaven oddělovač tisíců (musí být prázdný). Při práci nemusí být uživatel připojen k síti internet.

Uživatelské prostředí:

Uživatelské prostředí tohoto programu je velmi podobné tabulkovému procesoru Microsoft Excel. Tzn., že data jsou přehledně rozdělena do pomyslné tabulky, skládající se z řádků (označených číslicemi) a sloupců (označených písmeny). Z toho plyne jednoduchost práce s daty a snadná editace hodnot. Výhodou pro uživatele může být znalost práce s aplikací Excel, díky již zmíněné podobě uživatelského prostředí. Všechny funkce programu se volají pomocí menu, nebo lze využít horkých kláves pro rychlejší ovládání. Modul poskytuje krátkou nápovědu týkající se popisu programu WinGAp. Je možno v ní nalézt upozornění, informace o nastavení, výpočtu aproximace, výstupním textovém souboru, odchylkách apod.

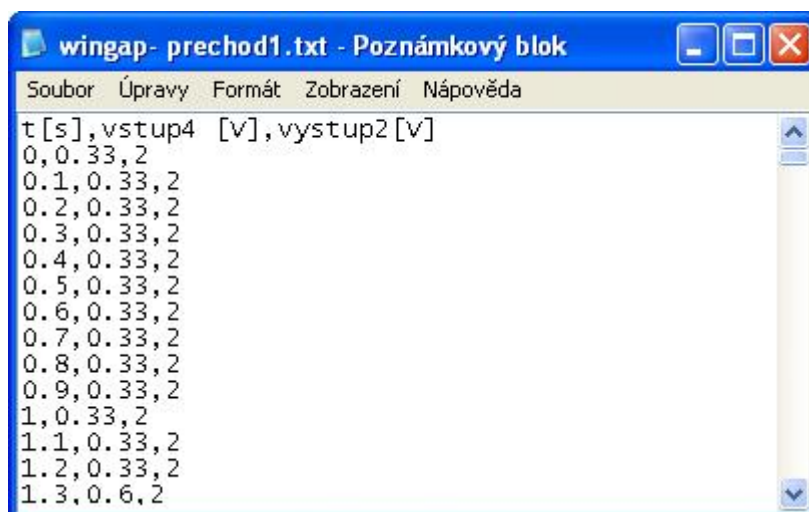


	A	B	C	D	E	F	G
1	t[s]	vstup4 [V]					
2	0	0,33					
3	0,1	0,33					
4	0,2	0,33					
5	0,3	0,33					
6	0,4	0,33					
7	0,5	0,33					
8	0,6	0,33					
9	0,7	0,33					
10	0,8	0,33					
11	0,9	0,33					
12	1	0,33					
13	1,1	0,33					
14	1,2	0,33					
15	1,3	0,6					

Obr. 4 Pracovní prostředí programu WinGap s načtenými daty

Vstupy a výstupy:

Hodnoty lze zadávat do pomyslné tabulky (viz obr. 4), přičemž první sloupec je vždy nezávisle proměnná a ostatní jsou závislé proměnné. Program tedy pracuje s jednou nezávisle proměnnou a více závisle proměnnými (maximálně však se sedmi, neboť celkově je možné využívat nejvíce osm sloupců) [BRODA, R. 2005]. Využití počtu řádků je omezeno na 3000. Program pracuje s formátem souboru označeným *.txt a nepodporuje více formátů.



```

t[s],vstup4 [V],vystup2 [V]
0,0.33,2
0.1,0.33,2
0.2,0.33,2
0.3,0.33,2
0.4,0.33,2
0.5,0.33,2
0.6,0.33,2
0.7,0.33,2
0.8,0.33,2
0.9,0.33,2
1,0.33,2
1.1,0.33,2
1.2,0.33,2
1.3,0.6,2

```

Obr. 5 Formát výstupu dat z programu WinGap

Data uložená v takovém formátu souboru jsou v textové podobě. Hodnoty se v něm ukládají do sloupců spolu s jejich názvy a jsou odděleny čárkou (viz obr. 5). Tyto názvy však neslouží jako popisy os (ty se do souboru neukládají vůbec), ale jako názvy proměnných. Při ukládání aproximovaných dat se do souboru ukládá nejprve nezávisle proměnná, poté závisle proměnná a nakonec odchylka aproximace.

Samotné grafy je možno uložit též ve formě obrázku a to s využitím 24 bitové mapy (formátu souboru *.bmp). Každý takto uložený graf (obrázek) má velikost 1024 x 768 obrazových bodů (pixelů), což neodpovídá velikosti okna grafu. Velice špatné je, že na obrázku jsou vidět pouze osy, vykreslené průběhy určitých dat, popř. jejich aproximace. Chybí zde jakékoli popisy os, jednotky, číselné hodnoty na osách, nadpis grafu a legenda, což dělá tyto obrázky nepoužitelné. Modul poskytuje také tisk, ale pouze pro grafy. Neumožňuje tisknout data, ať už zadané, načtené či aproximované.

Zpracování dat:

Aplikace umožňuje následující typy aproximace zadaných či načtených dat:

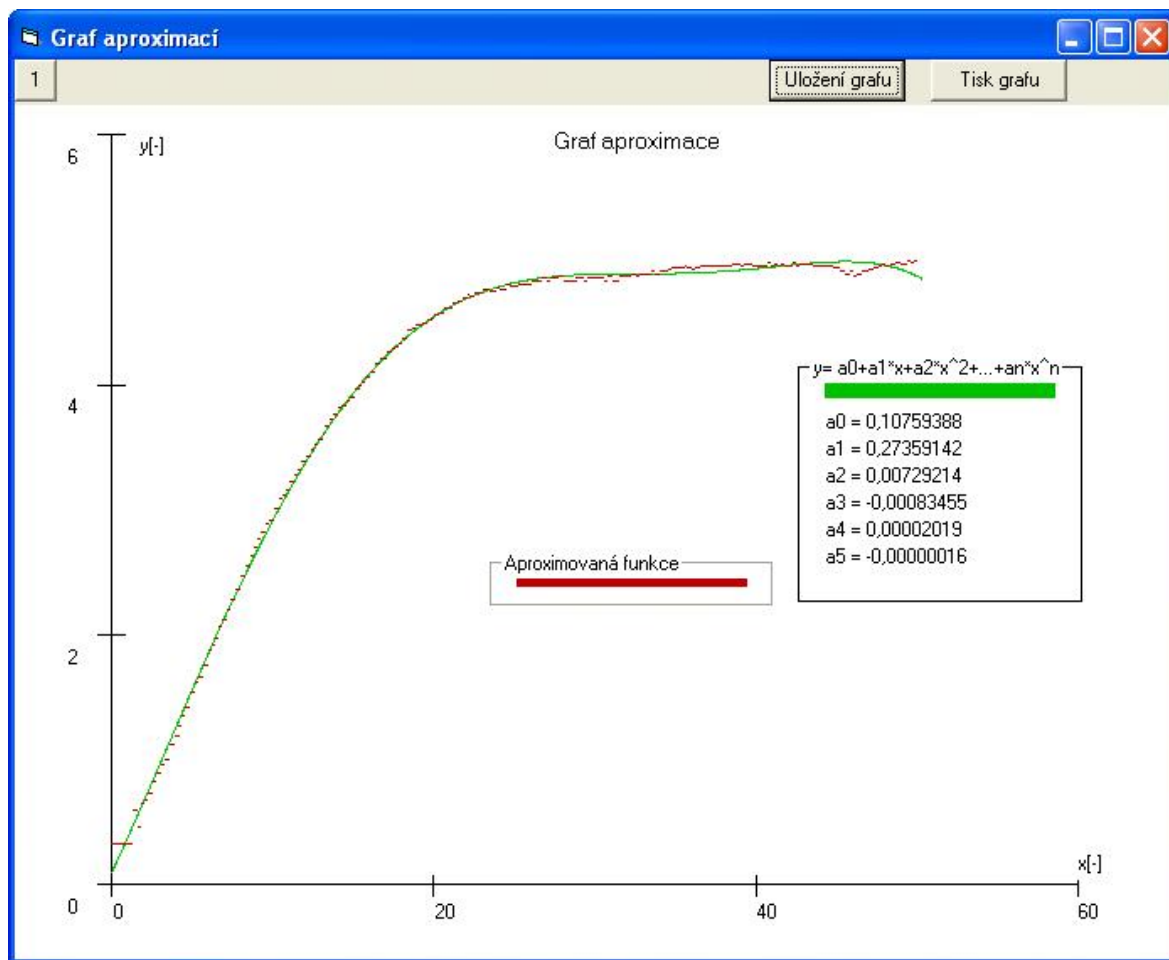
- přímka – volitelný počet úseků
- polynom $y = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$ pro $n \in \langle 1, 8 \rangle$
- polynom $y = a_0 + \frac{a_1}{x} + \frac{a_2}{x^2} + \dots + \frac{a_n}{x^n}$ pro $n \in \langle 1, 8 \rangle$
- funkce $y = a.e^{b.x}$
- funkce $y = a.\left(1 - \frac{1}{e^{\frac{x}{b}}}\right)$
- funkce $y = a.x^b$
- funkce $y = \frac{x}{a + b.x}$

Ačkoli se v prostředí modulu vyskytuje několik závislých proměnných, aproximovat lze vždy pouze jednu. V případě aproximace pomocí přímky zde oproti modulu APROX přibyla možnost aproximovat několika nebo maximálním počtem úseků [KOPECKÝ, M. 2003]. Pro aproximace pomocí polynomu má uživatel stejnou možnost volby stupně polynomu ($n \in \langle 1, 8 \rangle$).

Zobrazení dat:

Aplikace má k dispozici různé typy zobrazení dat. Je možno vykreslit graf zadaných či načtených dat buď se všemi závisle proměnnými, nebo jen s některými. Další možností je

pak zobrazení grafu aproximace se všemi, nebo jen s vybranými typy aproximace (jedné závislé proměnné). Také graf odchylek aproximace od výchozích dat lze vykreslit se všemi nebo jen se zvolenými typy aproximace. Menší nevýhodou je, že nelze vykreslit graf aproximace spolu s grafem odchylek. Naopak výhodou může být volba tloušťky vykreslovaných bodů či čar. Legendu grafu lze jednoduše přemísťovat na libovolné místo v grafu technikou **Drag & Drop** (táhni a pusť). Na obr. 6 je vidět měřená veličina v závislosti na čase (červeně) a její aproximace polynomem 5. stupně.



Obr. 6 Zobrazení načtených dat (červená) a jejich aproximace (zelená)

Na obrazovku je možné vypsát tabulku hodnot obsahující zadané či načtené údaje o závislé a nezávislé ose a odchylky. Dále parametry aproximace, jako jsou koeficienty aproximační funkce, index korelace, přiléhavost, střední odchylka a maximální odchylka.

Přes menu a nastavení se dá měnit tloušťka čáry pro vykreslení průběhů dat, popisky os včetně jednotek a nadpisy jednotlivých grafů. Grafy mají automatický rozsah hodnot, který nelze manuálně měnit. Uživatelsky definované nastavení programu je možné uložit do nebo načíst z konfiguračního souboru.

3.3 Programový modul JAP

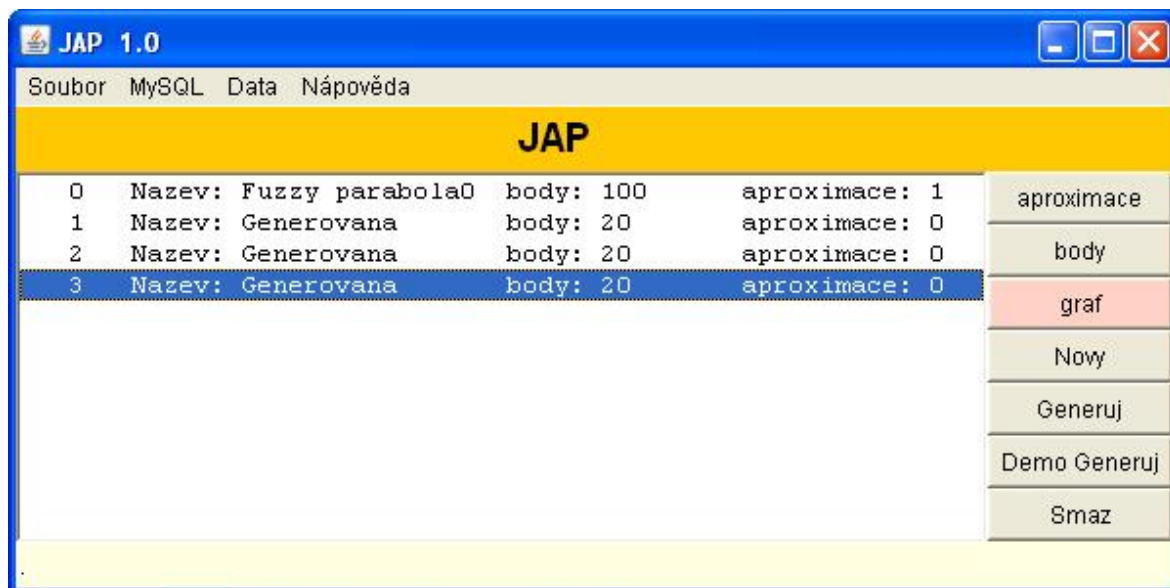
Tato aplikace slouží pro zpracování zadaných či experimentálně získaných dat a umožňuje jak jejich vizualizaci, tak i aproximaci a následnou prezentaci. Modul byl vytvořen pomocí programovacího jazyka Java. Výhodou je, že může být součástí HTML stránky jako *applet*, nebo je možné jej spustit jako klasickou aplikaci z pevného disku. Pro spuštění však vyžaduje grafický OS s podporou Javy ve verzi 1.4.1 nebo vyšší.

Systémové požadavky:

Program je volně dostupný a šířitelný. Nemá žádné licenční omezení. Nároky na hardware jsou určeny OS s GUI. Je také potřeba mít výkonný procesor pro numerické řešení aproximace metodou nejmenších čtverců. Výhodou jsou dokonce počítače s více procesory, procesory s více jádry nebo jejich kombinace, protože numerický výpočet umožňuje spuštění úlohy ve více vláknech současně [BRODA, R. 2005]. Program je vytvořen v jazyce Java, a proto je důležitá a zároveň nutná podpora Javy verze 1.4.1 nebo vyšší. Modul není potřeba instalovat, pouze podporu Javy. V případě online práce s aplikací v prostředí webového prohlížeče musí být uživatel připojen k síti internet. Pokud si však uloží program na disk, pak není připojení vyžadováno.

Uživatelské prostředí:

Prostředí modulu JAP je koncipováno do jednoho (hlavního) okna, v němž se nachází menu a také další ovládací panel v pravé části okna. Všechny funkce programu jsou tedy dostupné hned v prvním okně.



Obr. 7 Pracovní prostředí programu JAP s vygenerovanými daty

Aplikace nepodporuje obsluhu pomocí horkých kláves pro rychlejší ovládání. Umožňuje jen pohyb v menu pomocí šipek na klávesnici. Na druhou stranu grafy poskytují interaktivní ovládání myší (přiblížení a posun), což je uživatelsky příjemné. Program je vybaven obsáhlou nápovědou týkající se změn bezpečnostních práv a ovládání programu.

Vstupy a výstupy:

Modul JAP by měl být zpětně kompatibilní se staršími aproximačními moduly APROX a WinGAp (což není – viz níže). Umožňuje export dat do jejich formátu souboru, ale při ukládání je nutné zapsat správnou příponu souboru. Dále aplikace podporuje import a export dat pomocí univerzálního převaděče, kde je možné zvolit formát s pevnou šířkou sloupců nebo s oddělovači. Hodnoty lze obtížně zadávat ručně, nebo načíst ze souboru (funguje špatně, nebo vůbec). Modul by měl mít schopnost připojit se k databázi MySQL (neodzkoušeno), která slouží pro vzdálený přístup k datům (viz obr. 8). Otázka je, zda je to účelné.



Obr. 8 Okno pro připojení na MySQL databázi

Grafy je možné uložit ve formě obrázku v grafickém formátu souboru *.jpeg. Ten ale používá ztrátovou kompresi a proto není příliš kvalitní. Na obrázku grafu je vidět vše potřebné, stejně jako v okně grafu (tzn. nadpis, osy, názvy os, legenda, mřížka, průběhy apod.). Aplikace neposkytuje žádný tisk.

Zpracování dat:

Aplikace umožňuje následující typy aproximace zadaných či načtených dat:

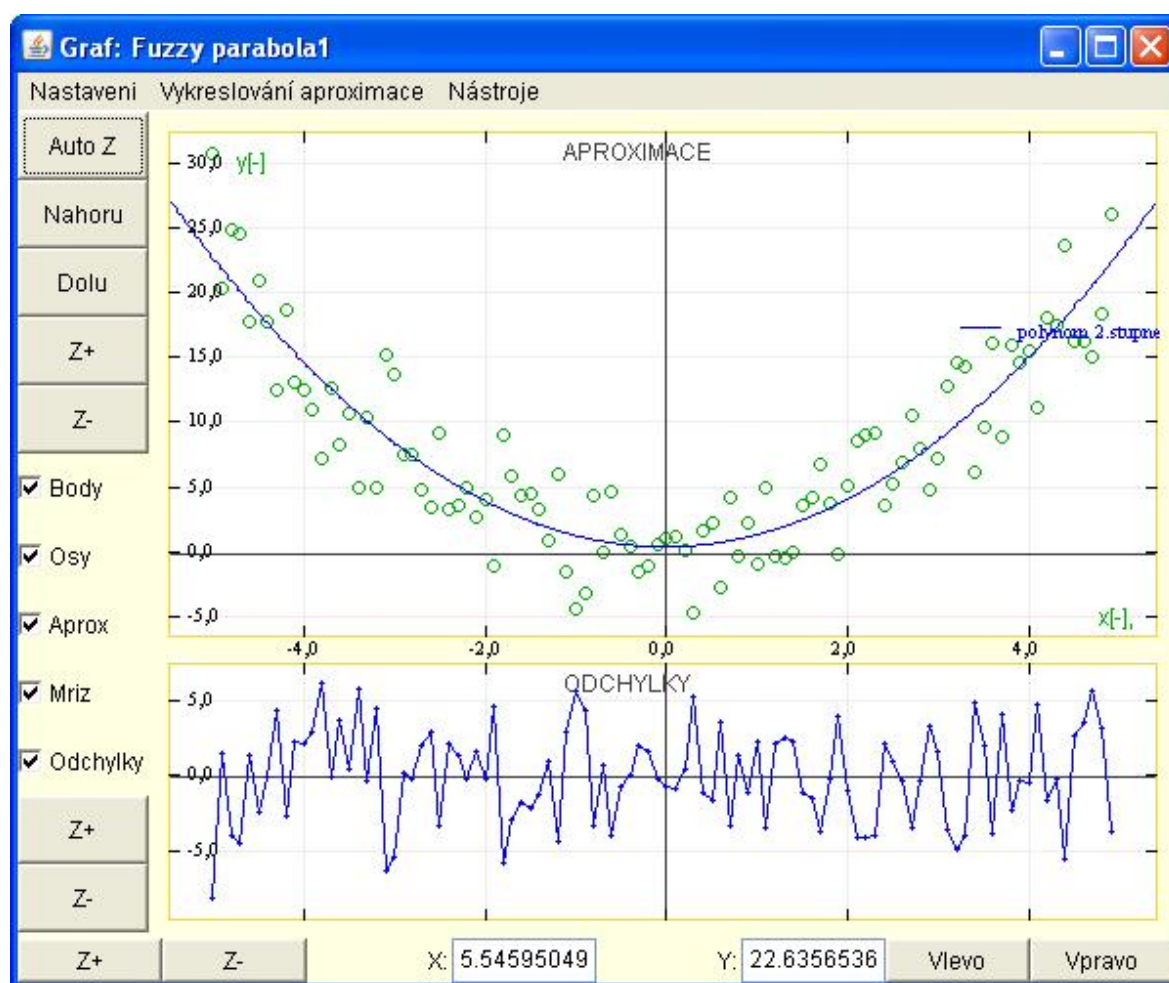
- analyticky – polynom s libovolnou volbou stupně polynomu
- kubická spline funkce
- numericky – funkce $a.x + b$
- numericky – funkce $\frac{1}{T_1 s + 1}$
- numericky – funkce $\frac{s}{T_1 s + 1}$
- numericky – funkce kmitavá $\frac{1}{T_0^2 s^2 + 2\xi T_0 s + 1}$

Zobrazení dat:

Modul umožňuje různé typy zobrazení dat. Grafy slouží pro zobrazení dat a aproximací, popř. i s odchylkami. Jsou schopny práce ve dvou režimech. První zobrazí jeden blok dat s jeho aproximacemi. A druhý režim zobrazí všechny bloky dat obsažené v programu. Do těchto režimů je možné se přepnout automaticky v závislosti na tom, odkud je volána inicializace grafu. Okno, kde se provádí vizualizace vstupních dat a jejich aproximací je tvořeno grafem s ovládacími tlačítky a menu. Výhoda je, že graf umožňuje interaktivní ovládání pomocí myši (přiblížení a posun). Obsah plochy grafu je možné uložit jako obrázek v grafickém formátu *.jpeg. Ten však díky použité ztrátové kompresi není příliš kvalitní.

Na obrazovku je možno vypsát tabulku dat obsahující údaje o závislé a nezávislé ose a odchylky. Také parametry aproximace, jako jsou koeficienty aproximační funkce, index korelace, přiléhavost, střední odchylka a maximální odchylka. Přes menu a nastavení lze měnit barva (osy, mřížky, plochy a okrajů), velikost bodů a šířka okraje okna od vykreslovaného grafu.

Na obr. 9 je uveden vykreslený graf z generovaného dema (tlačítko na hlavním panelu formuláře), neboť se nepodařilo správně provést import souboru, generování dat funguje „divně“ a zadávání hodnot ručně je nepohodlné a příliš zdlouhavé.



Obr. 9 Graf vygenerovaného dema

3.4 Shrnutí vlastností programových modulů

Programové moduly je poměrně lehké srovnat, neboť každé z řešení je dost odlišné od předchozího a to jak z pohledu uživatelského rozhraní aplikace a běhového prostředí, tak z pohledu stability, funkčnosti, praktičnosti a nabízených možností zpracování a prezentace dat. Aplikace APROX je stavěna pro MS-DOS a toto uživatelské rozhraní je dnes již nevyhovující. Zato program WinGAp, jak už z názvu plyne, pracuje pod Windows, který poskytuje GUI, což je určitý pokrok vpřed oproti verzi APROX. Navíc tento program poskytuje práci v tabulkovém procesoru, což je v případě velkého množství dat asi nejlepší možné pracovní prostředí. Programový modul JAP může být spouštěn jak z prostředí internetu, tak lokálně z pevného disku PC, což je určitá výhoda oproti programu WinGAp. Uživatelské prostředí tohoto programu je však mnohem složitější, nepřehledné a navíc nejsou odladěné některé funkce. Podstatné výhody a nevýhody analyzovaných programových modulů jsou shrnuty v tab. 1.

Tab. 1 Přehled podstatných výhod a nevýhod analyzovaných programových modulů

	APROX	WinGAp	JAP
Výhody	Graf aproximace spolu s grafem odchylek	Běhové prostředí Windows, tabulkový procesor	Běhové prostředí síť internet, graficky dobře zpracované
Nevýhody	Běhové prostředí DOS, jedna závisle proměnná, nestabilní prostředí	Nepoužitelný export grafu do obrázku, nestabilní prostředí	Složitě a nepřehledné ovládání, špatné zadávání hodnot, chybí kontrola validity!

Z pohledu stability se chová nejhůře APROX, protože při pohybu myši se aplikace zasekne. Lepší stabilitu poskytuje WinGAp, který se při testování zasekl jen párkrát, zřejmě díky chybě, kterou programátor zapomněl ošetřit. Při práci s modulem JAP nevznikly po celou dobu žádné potíže se stabilitou programu.

Co se týče nabízených funkcí pro zpracování a prezentaci dat, tak nejvíce možností poskytuje modul JAP. WinGAp a APROX obsahují stejné typy aproximace dat a neumožňují výpočet a vykreslení dat přechodových charakteristik podle zadaných parametrů. Obecné možnosti nabízené programovými moduly (tisk, export apod.) jsou propracovány přibližně na stejné úrovni.

4. Návrhy programového řešení

V této kapitole jsou rozebrány různé možnosti dostupných technologií pro vytvoření vhodného programového řešení nové aplikace. Jako základní rozdělení programů podle běhového prostředí je možné považovat programy za spustitelné pod určitým, graficky orientovaným operačním systémem, nebo v prostředí internetu.

4.1 *Architektura .NET*

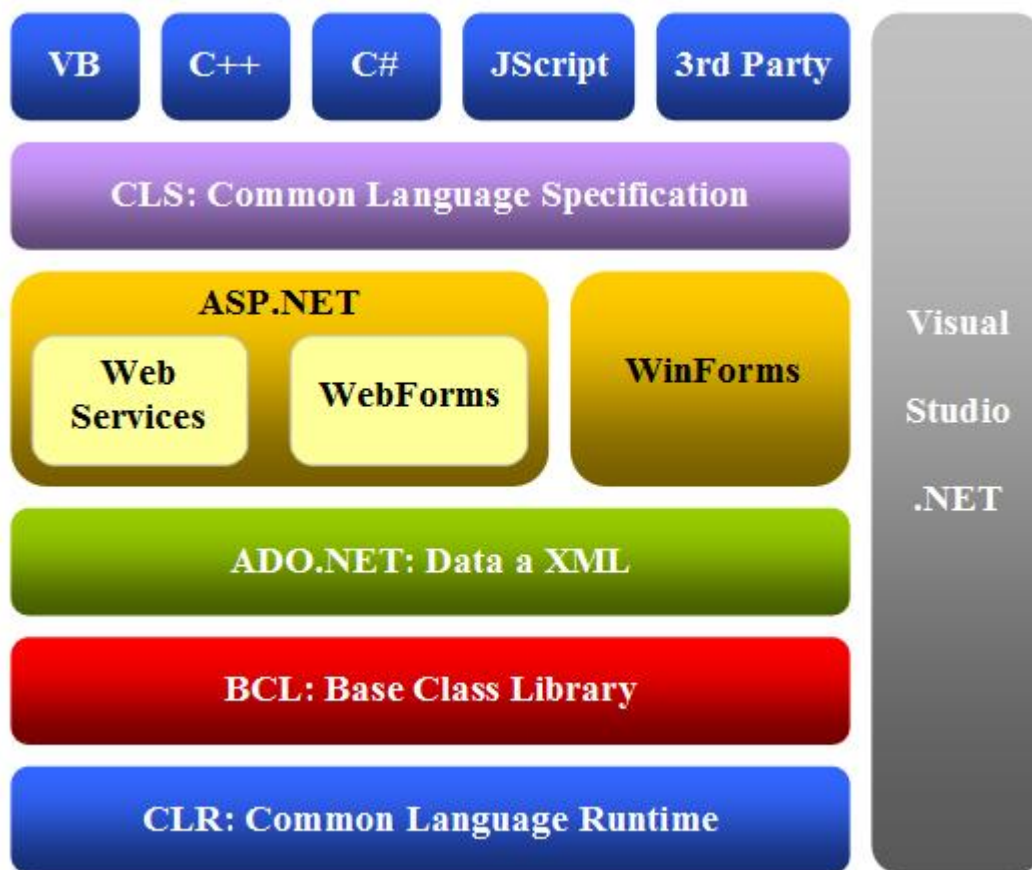
Pro každý programovací jazyk existuje speciální program, který se nazývá překladač – kompilátor. Ten zajišťuje překlad zdrojových kódů do instrukcí srozumitelných procesoru. Na platformě .NET překladač nekompiluje zdrojové kódy přímo do instrukcí procesoru, ale do jakéhosi zvláštního mezikódu, zvaného **IL** (*Intermediate Language*). Při spuštění programu je tento mezikód dopřeložen do konkrétních instrukcí daného procesoru. Díky tomu výkon aplikace nezávisí na zvoleném programovacím jazyku, ale na schopnosti použitého kompilátoru vytvářet rychlý a optimalizovaný mezikód. Prostor, ve kterém je mezikód vykonáván se nazývá **.NET Framework** a definuje rámec společných služeb či funkcí, dostupných v každé .NET aplikaci.

.NET Framework:

.NET Framework poskytuje společné prostředí – aplikační rámec – pro všechny .NET aplikace. Umožňuje následující základní úkony:

- Je běhovým (*run-time*) prostředím pro aplikace. Toto prostředí zajišťuje spouštění a provádění uživatelem vytvořených aplikací a služeb.
- Obsahuje knihovnu objektově orientovaných a jazykově nezávislých knihoven, které poskytují řadu služeb, jako jsou práce s databázemi (ADO.NET), vstup/výstup, procesy, vlákna, bezpečnost a komunikační protokoly.
- Umožňuje tvorbu webových služeb a nové generace ASP stránek.
- Poskytuje jednotnou architekturu pro tvorbu aplikací s klasickým či webovým uživatelským rozhraním.

.NET Framework zabezpečuje výše zmíněné úkoly. Základem .NET Frameworku je běhové prostředí nazývané **CLR** (*Common Language Runtime*). To zabezpečuje běh a poskytuje prostředí pro aplikace, které jsou vytvořeny s využitím tříd z knihovny **BCL** (*Base Class Library*). Třídy BCL mohou být např. pole, kolekce, manipulace se soubory, přístup k databázím, knihovny WinForms a WebForms.



Obr. 10 Microsoft .NET Framework

Common Language Runtime (CLR):

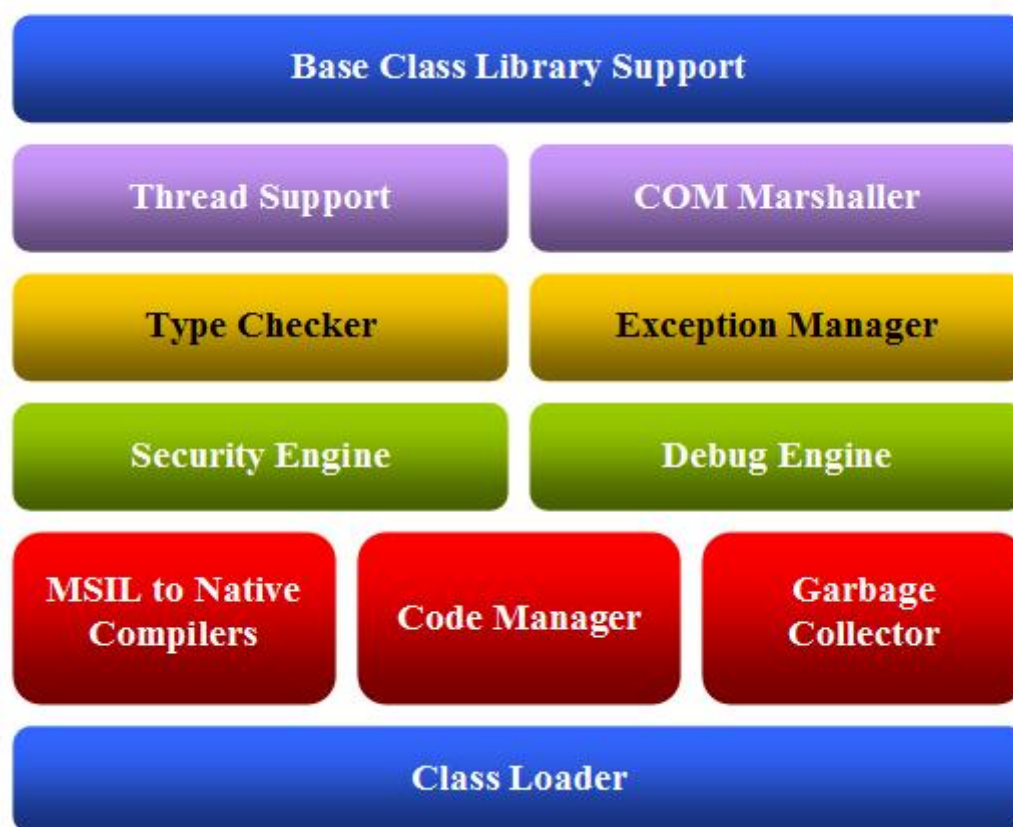
Common Language Runtime (CLR) má úkol zavést a vykonat mezikód IL. Výkon mezikódu IL je v prostředí .NET Framework nazýván řízeným kódem (*managed code*). Některé jazyky jako C# a C++ umožňují mix řízeného a neřízeného kódu (*unmanaged code*). Konkrétně VB.NET to neumožňuje, což je možná výhoda pro programátory začátečníky, neboť *managed code* je spravován běhovým prostředím a tím je zabezpečeno, že nedojde např. k chybám při správě paměti apod.

Hlavním důvodem zavedení IL je snaha o jednoduché přenášení existujícího kódu napříč hardwarovými platformami, jako jsou 32 bitové a 64 bitové procesory pro PC, pracovní stanice a servery, až po zařízení typu PDA apod. Tyto platformy jsou většinou hardwarově nekompatibilní a vyžadují tak rekompilaci kódu pro každou platformu zvlášť. Existence IL a CLR právě tento problém řeší.

CLR však v prostředí .NET Framework zajišťuje i jiné, stejně důležité úkoly:

- Zjednodušení tvorby a vývoje aplikací.
- Poskytnutí robustního a bezpečného prostředí pro běh aplikací.
- Podporu velkého množství programovacích jazyků.
- Ulehčení nasazení a správy aplikací.

Všechny výše zmíněné úkoly vyústili v implementaci sady *run-time* služeb, dostupných ve formě tříd a struktur implementujících různá veřejná rozhraní. Tuto množinu služeb je možné rozdělit do dvou kategorií. První obsahuje klasické aplikační funkce a služby prostředí Windows a druhá je doplňuje o sadu dalších služeb vyžadovaných moderními vývojovými postupy (př. *garbage collector* řídící životnost objektů v paměti, typovou kontrolu, podporu pro pokročilé ladění apod.).



Obr. 11 Komponenty CLR

Base Class Library (BCL):

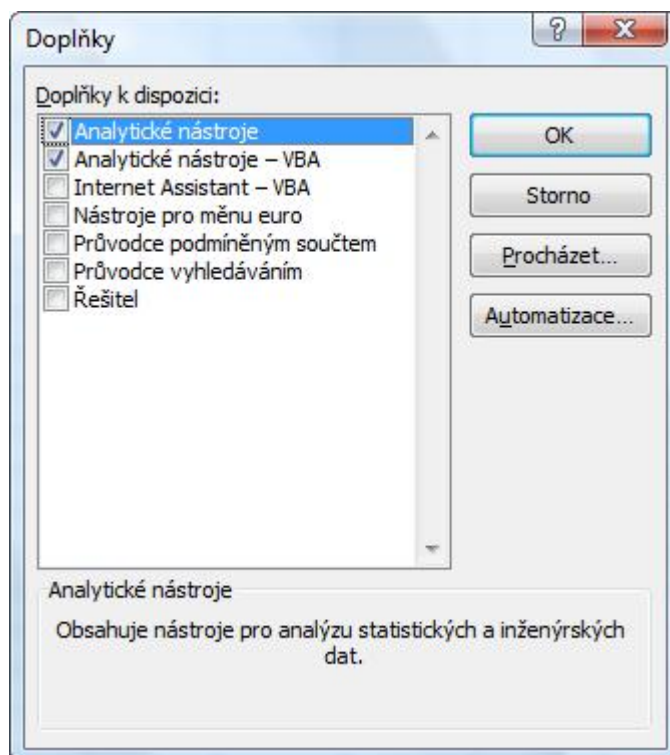
V historii došlo poprvé ke sjednocení rámce poskytovaných funkcí do jednoho balíku, který je k dispozici všem .NET kompatibilním jazykům. Tento balík se nazývá *Base Class Library* (BCL). Tzn., že ze všech těchto jazyků mohly být odstraněny funkce, které jsou sice v každém programu vyžadovány, ale odlišně nazývány v každém vývojovém nástroji.

Příkladem mohou být funkce pro manipulaci s řetězci, které jsou soustředěny do logických jednotek – tříd a jmenných prostorů.

Díky překladu zdrojového kódu do IL a použití společných knihoven BCL jsou všechny .NET kompatibilní jazyky postaveny na stejnou výkonnostní úroveň. Více je možno dozvědět se např. v [KAČMÁŘ, D. 2001, LACKO, L., MICROSOFT 2005].

4.2 Doplněk pro MS Excel – VB.NET

Vytváření doplňků má pro vývojáře v prostředí Microsoft Excel velký užitek. Doplněk si lze představit jako něco, co se přidá do softwaru z důvodu zvýšení jeho funkcionality. MS Excel obsahuje i několik vlastních doplňků, zahrnujících nástroje analýzy dat apod. (viz obr. 12).



Obr. 12 Doplňky aplikace MS Excel

Optimálně se nové možnosti a funkce v podobě doplňku zakomponují do programu tak, že se jeví jako součást programu. Přístup Excelu k vytvořeným doplňkům je docela dobrý, protože jakýkoliv uživatel si jej může vytvořit sám ze sešitu. Doplněk je v podstatě jinou formou klasického sešitu (pokud se jedná o doplněk na úrovni dokumentu). Jakýkoli sešit může být převeden na doplněk, ale ne každý je k tomu vhodný. Doplněk odlišuje od klasického sešitu přípona souboru ***.xlam**. Doplňky jsou vždy skryté a není tedy možno sešit, list, či graf obsažený v doplňku spustit. Lze k němu však přistupovat přes procedury programovacího jazyka *Visual Basic for Application* (VBA) nebo VB.NET a zobrazovat

různá dialogová okna, která jsou součástí uživatelských formulářů. Více informací je možno najít v [WALKENBACH, J. 2007], nebo také v [ETHERIDGE, D. 2007].

Obecně lze doplněk tvořit dvěma způsoby. Buďto přímo v Excelu pomocí VBA a bez jakýchkoliv dalších nástrojů (jak je popsáno výše), nebo také ve Visual Studiu jako profesionální doplněk s využitím možností platformy .NET a jazyka VB.NET. První varianta má výhodu v tom, že takovéto řešení je velmi jednoduché, avšak ne zcela ideální. U druhé varianty je jasné plus využití všech výhod .NET Frameworku.

Dále je možné doplňky rozčlenit do dalších dvou skupin a to podle toho, jak byly vytvořeny. Existují totiž doplňky na úrovni dokumentu, nebo na úrovni aplikace (tzv. COM doplňky). Způsob na úrovni aplikace je lepší, neboť se využívá častěji a více se doporučuje. Takovýto doplněk se implementuje do celé aplikace a není propojen jen s jedním dokumentem či sešitem. Používá se většinou pro rozsáhlejší doplňky. Má jednu nevýhodu, že se musí na daný počítač instalovat jako klasický program. Funguje tak, že když se spustí Excel, objeví se všechny nové ikony, tlačítka a nástrojové panely.

Jednodušší způsob je sice dělat doplněk přímo v Excelu, ale s použitím Visual Studia a platformy .NET lze dosáhnout větších možností a aplikace jsou pak mnohem profesionálnější, což je určitě výhoda. I přes to, že lze ve VBA naprogramovat téměř všechno, spousta věcí se musí dělat složitě přes WIN32 API.

Jediná nevýhoda doplňku obecně je závislost na aplikaci Excel, ale zase zde odpadá problém s grafy. Obrovskou výhodou je přehledné a známé pracovní prostředí tabulkového procesoru a výborné grafy.

4.3 Aplikace WIN32 – VB.NET

Samostatný program v prostředí Windows by pro účel této práce měl podobu MDI aplikace, která má vlastní, hlavní okno, chovající se jako rodič (*parent*), obsahují podokna neboli potomky (*child*), jež poskytují informace o datech, aproximacích a obsahují vykreslené grafy. Je to další možná varianta řešení zpracování a prezentace experimentálních datových souborů. Takovýto způsob je považován za hodně profesionální, ryze vlastní a originální řešení. Projekt je však značně náročnější a vyžaduje speciální komponentu na kreslení grafů. Tu je možno buďto naprogramovat (což je časově náročné), nebo zakoupit od třetích stran.

Další možností samostatné WIN32 aplikace je využít komponentu pro grafy z programu Excel, jako tomu je např. u programu **Signal Analyzer** od prof. Ing. Jiřího Tůmy, CSc. Při využívání komponenty grafu z Excelu je ale nutné, aby byla provedena

instalace produktu Excel předtím, než se komponenta začne používat. Takovýto způsob řešení je sice jednoduchý, ale neobvyklý a vede opět k programu Excel, z čehož plyne, že je lepší zvolit buďto první variantu a tvořit doplněk přímo pro Excel anebo zvolit úplně jiné řešení. Oproti tomu samostatná aplikace bez využití Excel komponenty pro grafy nepotřebuje využívat nic jiného. Bylo by však těžké nahradit kvalitní prostředí tabulkového procesoru aplikace Excel.

4.4 Serverová aplikace ASP.NET – VB.NET

ASP.NET je nástupce předchozí technologie ASP. Je založen na *Common Language Runtime* (CLR), který je sdílen všemi aplikacemi postavenými na .NET Frameworku. Největší rozdíl mezi těmito technologiemi je v tom, že kódy v ASP.NET stránkách jsou kompilované. Tím pádem vzniká kompilovaný kód, který je mnohem rychlejší. ASP.NET stránky potřebují pro svůj provoz webový server a samozřejmě platformu .NET Framework, což plyne už z názvu ASP.NET. Mezi některé výhody této technologie patří např. již zmíněná rychlost aplikací díky předkompilovanému kódu, objektové programování, možnost využití spousty programovacích jazyků (VB, C++, C#, Jscript, J#, Perl, Python, ...) a skutečnost, že konfigurační údaje jsou uloženy v XML souborech [LACKO, L.].

U tohoto řešení, které se může zdát výhodné, výjimečně není prospěšný přístup k aplikaci přes internet. Ta má totiž prioritně sloužit studentům v laboratořích katedry Automatizační techniky a řízení, kde není zapotřebí přistupovat k aplikaci z prostředí webu. Navíc tato mnohdy označovaná výhoda s sebou nese zároveň nevýhodu a to nutnost připojení k síti internet. ASP.NET webové stránky se tvoří i programují ve Visual Studiu podobně, jako běžné Windows aplikace. Komponentu pro grafy je možno rovněž naprogramovat anebo stáhnout některou z free verzí grafových komponent z internetu. Možnosti takových komponent jsou např. vykreslování grafu přímo do webové stránky, nebo poslání dat v XML formátu do Adobe Flash, který následně data vykreslí (většinou v hezčí podobě). Nicméně z tohoto řešení lze jen velmi obtížně exportovat vykreslený graf do aplikace Microsoft Word, což je také jeden z důležitých požadavků na nový programový modul.

4.5 Shrnutí možností programových řešení

Jak už bylo v předchozích podkapitolách naznačeno, každé řešení má své určité výhody, ale i nevýhody. Z několika nastíněných možností se jeví jako jednoznačně nejlepší vytvořit doplněk pro aplikaci Microsoft Excel, pomocí jazyka VB.NET na aplikační úrovni, s využitím všech výhod platformy .NET a programování v příjemném prostředí Visual Studio s nástrojem *Visual Studio Tools for Office System 2007*. Toto řešení obsahuje nejvíce výhod. Zároveň má jednu výjimečnou výhodu, a sice pokud uživateli nebude doplněk poskytovat potřebné nástroje, nemusí opouštět toto prostředí, ale může využít běžných nástrojů programu Excel.

Výhody řešení doplňku aplikace Excel uvedené v bodech:

- Prostředí Excel je naprosté většině uživatelů PC známé a poskytuje přehlednou a rychlou práci s daty
- Výborná možnost exportu grafů do Wordu
- Intuitivní a přehledné ovládání programu
- Nevyžaduje připojení k síti internet
- V případě, že si uživatel nevystačí s doplňkem, může bez problémů využít běžné funkce Excelu
- Využití technologie *ClickOnce* pro nasazení doplňku (v případě umístění instalačního balíčku na webový server je možno provádět okamžité aktualizace)

Nevýhody řešení doplňku aplikace Excel uvedené v bodech:

- Pro plnou funkčnost vyžaduje Excel a Word ve verzi 2007 a .NET Framework 3.5
- Nutnost instalace doplňku

S konkrétním přihlédnutím na skutečnost, že program bude využíván především v laboratořích katedry Automatizační techniky a řízení, lze říci, že navržené programové řešení nemá žádnou nevýhodu. Sada nástrojů Microsoft Office je na počítačích v laboratořích vždy nainstalována (a taky se často využívá) a samotná instalace doplňku není nijak obtížná, ani časově náročná. Navíc ji lze provádět z prostředí webu. Co se týče instalace doplňku, tak jen poznamenám, že i u webových řešení je většinou potřeba něco doinstalovat. Např. *Flash Player* či podporu *Javy*.

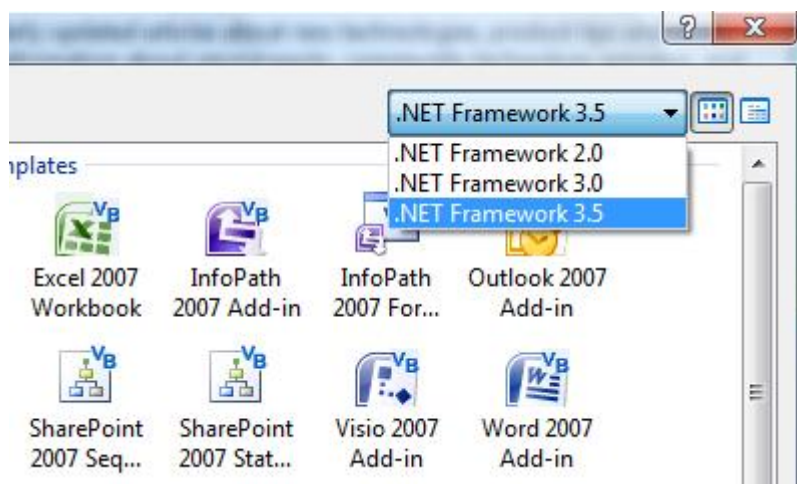
Zajímavostí instalace doplňku je také fakt, že pokud již počítač obsahuje příslušný .NET Framework, instalace nevyžaduje zásah administrátora, ani potřebu zvýšených přístupových práv.

5. Visual Studio 2008

Visual Studio je balík softwarových produktů, pokročilých vývojových nástrojů, funkcí pro práci s databázemi, ladicích funkcí, nástrojů a technologií pro rychlou tvorbu různých typů aplikací.

Oblibu si u programátorů získalo v roce 1995, ve verzi 6, kdy se díky doformovanému jazyku Visual Basic 6 otevřely možnosti i ne moc zkušeným programátorům. V roce 2001 přišlo s platformou .NET Framework verze 1, v roce 2005 vyšla verze .NETv2, v roce 2006 pak verze .NETv3.0 a současná verze .NET Frameworku, použitá ve Visual Studiu 2008 je verze .NET Framework 3.5 (SP1).

Novinkou ve Visual Studio 2008 je kromě poslední verze .NET Frameworku možnost zachování kompatibility se staršími verzemi .NET Frameworku. Výběr platformy se provádí při zakládání projektů (viz obr. 13), na což Visual Studio 2008 reaguje zpřístupněním pouze určitých projektových šablon a technologií [HANÁK, J.].



Obr. 13 Výběr platformy .NET Frameworku při založení projektu

Další novinkou je vylepšení vizuálního návrháře pro rychlejší vývoj s využitím .NET Framework 3.5 a také třeba to, že lze Visual Studio instalovat „side-by-side“. Znamená to, že je možné jej instalovat i v případě, pokud jsou na disku jiné verze Visual Studia.

5.1 Visual Studio Tools for Office

Nástroj *Visual Studio Tools for Office* (dále jen VSTO) existuje již dlouho. První verze, VSTO 2003, byla jako doplněk, který se musel doinstalovat do Visual Studia nebo bylo možné jej zakoupit samostatně jako separátní edici. Tento doplněk umožňoval vytvářet na bázi *Code-behind* kódu rozšíření pro aplikace Excel 2003 a Word 2003 na

úrovni dokumentů. Následující verze VSTO 2005 již umožňovala plnohodnotné využití .NET vývoje pro aplikace Excel 2003, Word 2003 a nově také pro Outlook 2003 a InfoPath 2003. Byla zde možnost vytvářet vlastní podokna úloh (tzv. *Actions Pane*) na úrovni dokumentů, dále *Host Controls* pro rozšíření Excel či Word dokumentů. Další, momentálně předposlední verze byla verze VSTO 2005 SE (*second edition*), což bylo jakési rozšíření verze 2005 o nově vzniklý Office 2007. Tato verze přinesla zároveň možnost vytváření doplňků na úrovni aplikace, vytváření vlastních Ribbon prvků, nebo podoken úloh na úrovni aplikace. Pokud by chtěl tedy programátor vyvíjet doplňky na úrovni dokumentů, použil by verzi VSTO 2005, pokud na úrovni aplikace, zvolil by verzi VSTO 2005 SE. Toto však nebylo ideální řešení, protože pokud chtěl programátor vytvořit podokno úloh na úrovni dokumentu, použil verzi 2005. Vytvořený dokument pak musel otevřít v Office 2007 a celý přemigrovat. VSTO 2005 SE umožňovalo rozšíření aplikací Excel 2007, Word 2007, PowerPoint 2007, Outlook 2007, Visio 2007 a InfoPath 2007 [NEUWIRTH, M. 2006, NEUWIRTH, M. 2008].

Současná a tedy poslední aktuální verze je VSTO 2008. Je součástí Visual Studia 2008 a přináší možnosti všech předchozích verzí. To znamená, že obsahuje podporu pro Office 2003 i Office 2007, umožňuje tvorbu doplňků a vlastních podoken úloh na úrovni aplikace i na úrovni dokumentu a nově zde vznikl tzv. návrhář (*designer*) pro tvorbu grafického prostředí Ribbon prvku. Ten bylo možné v předchozí verzi VSTO tvořit jen a pouze pomocí XML dokumentu.

VSTO 2008 podporuje tvorbu doplňků pro Office 2003 na úrovni aplikace pro:

- Excel 2003
- Word 2003
- PowerPoint 2003
- Outlook 2003
- Visio 2003

a na úrovni dokumentu pro:

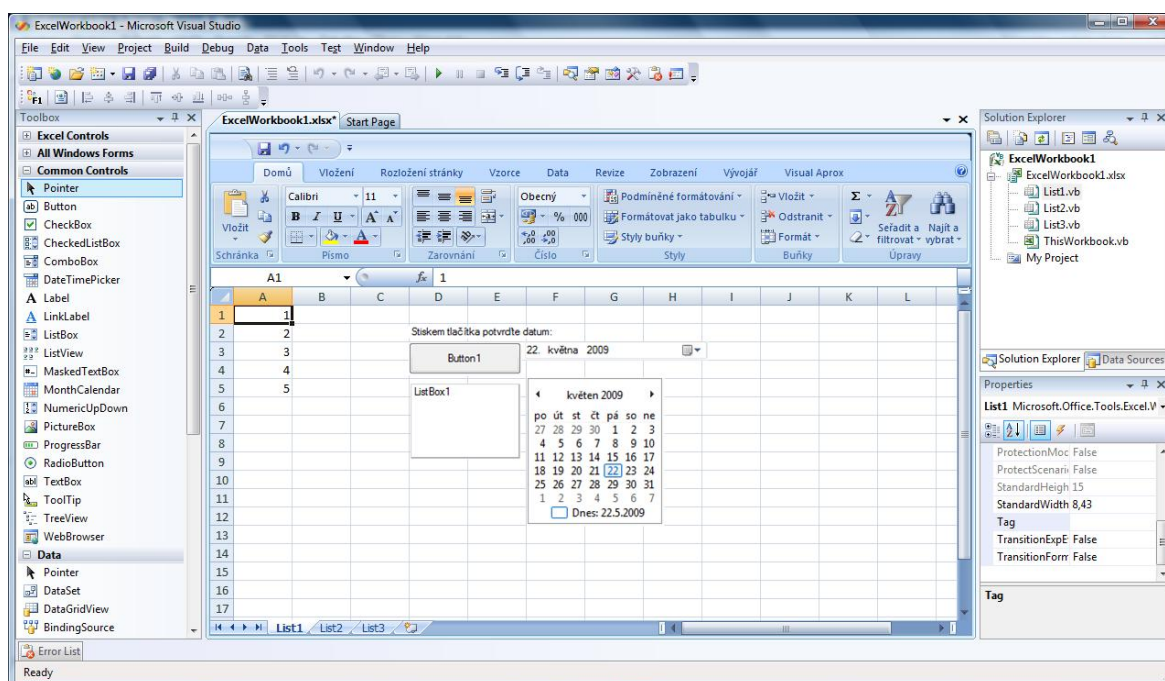
- Excel 2003
- Word 2003

VSTO 2008 však poskytuje mnohem větší možnosti pro Office 2007. Je možné vytvářet doplňky na úrovni aplikace, vlastní podokna úloh a libovolně upravovat stávající popř. tvořit nové Ribbon prvky, viz tab. 2.

Tab. 2 VSTO 2008 poskytuje tyto možnosti pro Office 2007

	Doplňěk na úrovni aplikace	Vlastní podokno úloh	Úpravy Ribbon prvku
Excel 2007	OK	OK	OK
Word 2007	OK	OK	OK
Outlook 2007	OK	OK	OK
PowerPoint 2007	OK	OK	OK
InfoPath 2007	OK	OK	---
Visio 2007	OK	---	---

Mezi další novinky VSTO 2008 se řadí např. poskytnutí možnosti řešení na úrovni dokumentu pro Excel, Word a nově také pro InfoPath. K dispozici jsou řízené ovládací prvky, které je možné umístit přímo na plochu dokumentů. Lze vytvářet vlastní podokna úloh, která jsou spojena přímo s daným dokumentem. Novinkou je také integrované prostředí Office přímo do prostředí Visual Studio 2008 viz obr. 14.

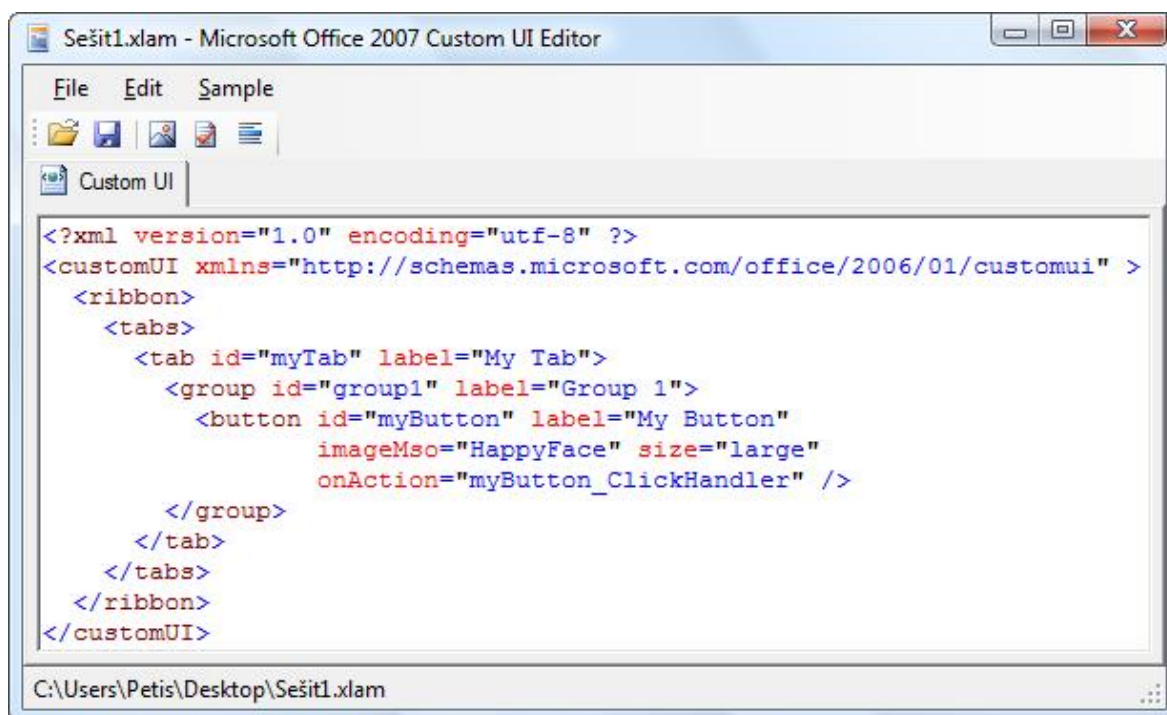


Obr. 14 Integrované prostředí Excelu přímo do Visual Studio

Pokud chce vývojář tvořit design dokumentu (v případě doplňků na úrovni dokumentu), tak se daný dokument otevře přímo v prostředí Visual Studio 2008 a není potřeba toto prostředí opouštět.

Další novinkou je např. návrhář pro tvorbu regionů pro Outlook 2007. Doposud byla možnost tvořit tyto regiony pouze pomocí návrháře, který je integrovaný v Outlooku [NEUWIRTH, M. 2008].

Nově je zde ještě interoperabilita mezi kódem napsaným ve VBA a kódem v .NETu pomocí VSTO. To znamená to, že je možné z VBA kódu zavolat kód napsaný v .NETu. Návod, jak přizpůsobit Ribbon prvek Excelu 2007 použitím VBA kódu je možno najít v [ROSENBLUM, M. 2007]. Pro lepší práci s XML dokumentem mimo Visual Studio je vhodné použít nástroj *Custom UI Editor*, jehož pracovní prostředí je zobrazeno na obr. 15. Více je možno dozvědět se taky v [WALKENBACH, J. 2007, STR. 301].



Obr. 15 Custom UI Editor

Automatizace Office aplikací se uskutečňuje za přispění *Primary Interop Assemblies* (PIA) sestavení (viz obr. 16), která formují rozhraní mezi řízeným a nativním (COM) světem [HANÁK, J.].

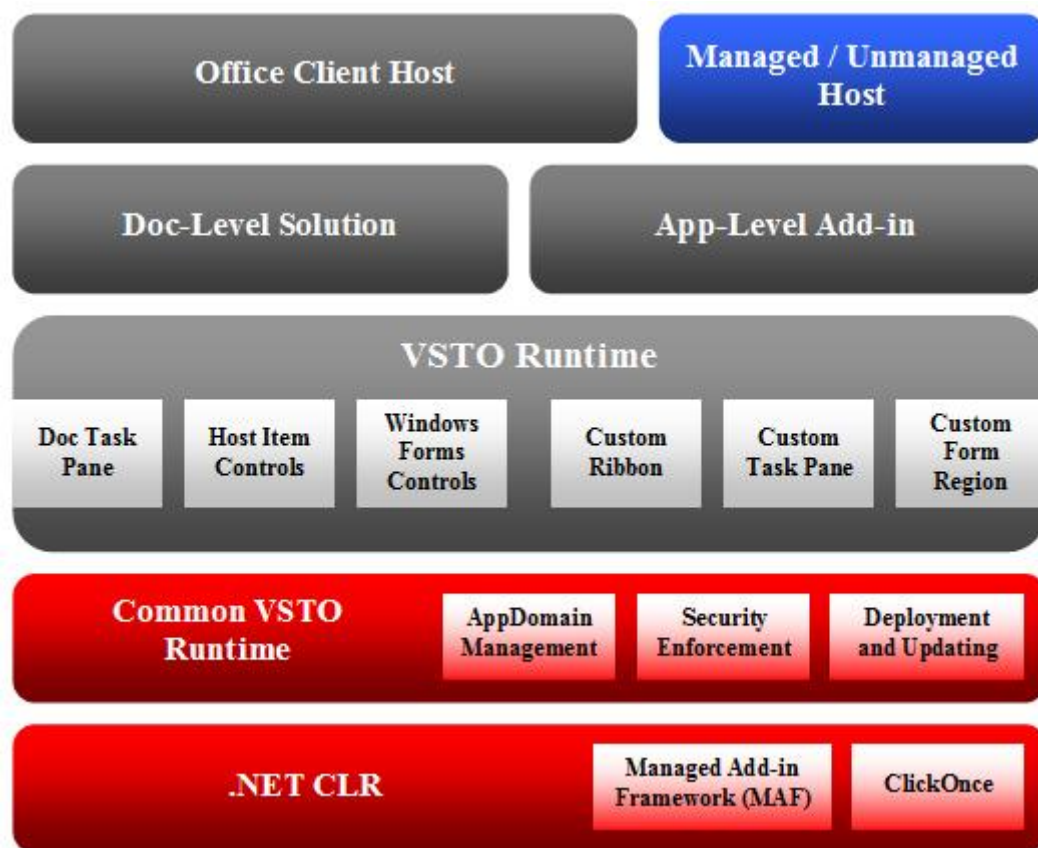


Obr. 16 Zařazení Primary Interop Assemblies

5.2 Architektura VSTO 2008

V podstatě nejdůležitější součástí VSTO je tzv. **běhové prostředí**, nebo také **VSTO Runtime**. Vlastní Runtime obsahuje dvě klíčové části. Na jedné straně je .NET Framework, který umožňuje nasadit dané řešení pomocí technologie *ClickOnce* a který obsahuje i *Managed Add-in Framework* (MAF). Na straně druhé je vlastní Office aplikace, která vytváří *Office Client Host Controls*, pomocí kterých je .NET připojen do COM prostředí. Nad vrstvou .NET CLR je obecná část VSTO Runtime, která spravuje aplikační domény, zajišťuje nasazení, aktualizace a další věci. Poté je samostatný VSTO Runtime, který obsahuje .NET objekty pro tvorbu doplňků, dokumentových řešení apod.

U řešení na úrovni dokumentu jsou k dispozici *Task Pane* pro dokumenty, *Windows Forms* ovládací prvky nebo *Word Content Controls* a u doplňků na úrovni aplikace to může být například *Custom Ribbon*.



Obr. 17 VSTO Runtime

Objekty VSTO:

- Host Item Controls
 - Microsoft.Office.Tools.Excel.ListObject
 - Microsoft.Office.Tools.Word.Bookmark
 - A další ...
- Windows Forms Controls
 - Microsoft.Office.Tools.Excel.Controls.Button
 - Microsoft.Office.Tools.Word.Controls.ComboBox
 - A další...
- Speciální ovládací prvky
 - Document Task Pane
 - App-Level Custom Task Pane
 - Custom form region

Oproti předchozím verzím VSTO je nyní k dispozici nástroj zvaný **ClickOnce**, pomocí kterého lze velice jednoduše nasazovat vlastní řešení. Tato technologie je stejná jako při nasazování standardních Windows aplikací pomocí této technologie. Vytvoří se jednoduchý instalační balíček, který obsahuje vše potřebné. Jedná se o instalaci pro jednotlivé uživatele. *ClickOnce* technologie a vlastní řešení nasazené touto technologií pak podporují aktualizaci a odinstalaci, aniž by bylo potřeba nějakého zvláštního zásahu uživatele. K dispozici je i podpora offline řešení.

Nejlepší na této technologii je, že nevyžaduje nástroj *Caspol* pro nastavení bezpečnosti. Všechny, nebo většina Office doplňků totiž vyžaduje větší bezpečnostní oprávnění nebo *full trust*. Při nasazování doplňků v předchozích verzích se muselo buď administrátorským zásahem, nebo nástrojem *Caspol*, popř. politikami nastavovat bezpečnostní nastavení. Pokud se však využije technologie *ClickOnce*, pak není potřeba nástroj *Caspol* a tato technologie již zajistí všechna potřebná oprávnění. Zkráceně lze říct, že se *ClickOnce* technologie postará i o bezpečné zavedení aplikačních doplňků [NEUWIRTH, M. 2008].

Hotový doplněk se **publikuje** pomocí *Publish Wizard*, který je součástí *Visual Studio* 2008. Řešení je možno publikovat na webový server, sdílené síťové prostředky, nebo třeba na CD či USB disk.

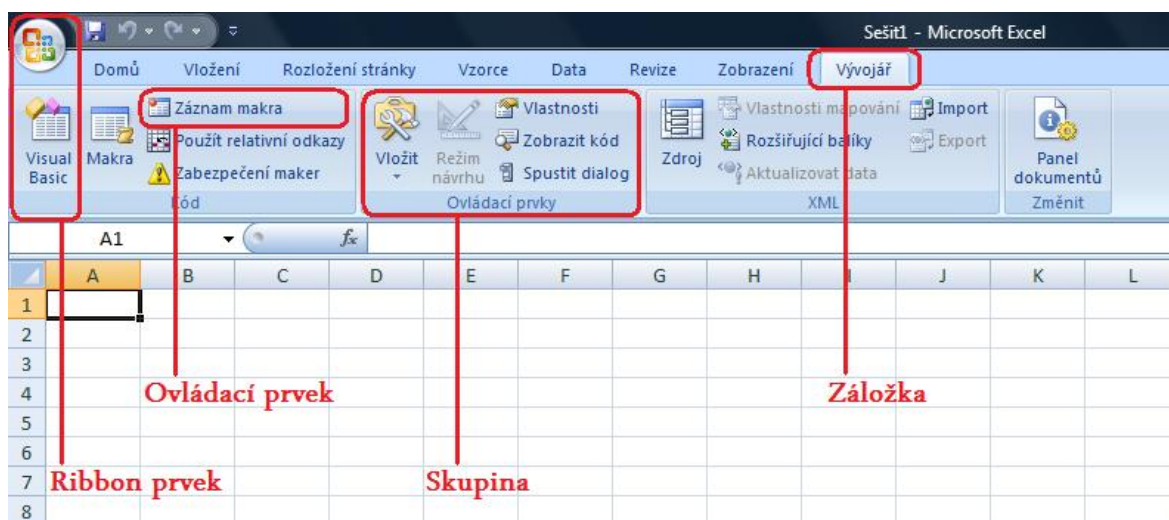
Instalace probíhající na straně uživatele vyžaduje, aby bylo k dispozici běhové prostředí. Pokud není, instalace sama nainstaluje běhové prostředí a VSTO Runtime, ale je vyžadován administrátorský účet, nebo zásah administrátora. V případě, že se na počítači

již vyskytuje běhové prostředí, instalace probíhá bez zásahu administrátora. Součástí instalace je registrace doplňku, konfigurace bezpečnosti a další. Pokud je vyžadována instalace běhového prostředí nebo VSTO Runtime bez zásahu administrátora, je možné využít např. služeb SMS [NEUWIRTH, M. 2008].

V případě nasazení vlastního řešení technologií *ClickOnce*, je možné určit, po kolika dnech má probíhat kontrola **aktualizace** (výchozí hodnota je nastavena na kontrolu po sedmi dnech). Stejně tak lze kontrolovat, zda už je k dispozici nová verze doplňku. Dále je možné nastavit, zda uživatel smí používat starší verze doplňku, nebo musí používat vždy novou, aktualizovanou verzi. Součástí je i offline *cache* v případě, že by řešení využívalo nějaké síťové zdroje. Další informace je možné najít např. v [BRUNEY, A. 2006, ANDERSON, T. 2009].

6. Ribbon prvek

Je to ovládací prvek vyvinutý společností Microsoft. Poskytuje nové uživatelské rozhraní v Office 2007. Jak je vidět na obr. 18, tento prvek je přehledně uspořádán do záložek, kde v každé záložce jsou určité skupiny ovládacích prvků, obsahujících tlačítka, popisky, či jiné ovládací prvky.



Obr. 18 Vzhled a popis Ribbon prvku

Úpravy tohoto prostředí se provádí pomocí XML dokumentu anebo nově také pomocí Ribbon Designeru (v prostředí *Visual Studio 2008* s využitím nástroje *Visual Studio Tools for Office* a *.NET Frameworku 3.5*). Úpravy lze provádět i mimo Visual Studio, pomocí XML dokumentu a jazykem VBA volat kód napsaný v *.NETu*. Toto už však bylo popsáno výše, viz str. 34.

VSTO 2008 podporuje úpravy Ribbon prvku v aplikacích Excel 2007, Word 2007, PowerPoint 2007 a Outlook 2007. Starší řešení, programované pro Office 2003 se v Ribbon prvku zobrazí následujícím způsobem (viz tab. 3).

Tab. 3 Možnosti zobrazení staršího řešení v Ribbon prvku

Kód, který přidává:	Zobrazen ve skupině:
Prvek do standardního menu	Menu Commands
Prvek do standardního toolbar	Toolbar Commands
Vlastní toolbar	Custom Toolbars
Skupiny se zobrazí v záložce Doplněk (Add-Ins)	

6.1 Možnosti přizpůsobení a ovládací prvky

V rámci Ribbon prvku existuje široká škála možností přizpůsobení a to zejména přidávání vlastních záložek, vkládání vlastní skupiny do již existujících záložek, přidávání ovládacích prvků do již existujících skupin. Je možné také přidávání vlastních položek do *Menu File* a přidávání položek do stávajících kontextových menu. Ne všechny možnosti jsou však obsaženy v Ribbon Designeru. Např. skrytí skupin a určitých ovládacích prvků je možné provádět pouze pomocí XML dokumentu. V následující tabulce (viz tab. 4) jsou uvedeny možnosti práce s Ribbon prvkem pomocí nového Ribbon Designeru a pomocí XML dokumentu.

Tab. 4 Tabulka možností editace Ribbon prvku v design módu a pomocí XML dokumentu

Vlastnost	Visual Designer	XML
Tvorby vlastních záložek	OK	OK
Úprava stávajících záložek	OK	OK
Skrytí záložek	OK	OK
Skrytí skupin a prvků	---	OK
Start from Scratch	OK	OK
Položky v Office Menu	OK	OK
Úpravy Quick Access Toolbar	---	OK
Úpravy Built-in Commands	---	OK

Kontextové nabídky v Ribbon prvků fungují dvojím způsobem. Standardní kontextové nabídky známé z předchozích verzí Office, kdy se kontextová nabídka objeví na ploše a druhá verze je, kdy jsou některé kontextové nabídky zobrazeny přímo v rámci Ribbon prvku jako speciální záložka a objeví se jenom v případě, kdy se pracuje s určitými prvky v rámci dokumentu. Příkladem takovéto kontextové nabídky může být, když uživatel v rámci aplikace Excel klikne na graf (ten se stane aktivní) a v Ribbon prvků se objeví další, speciální kontextová nabídka, která tam předtím nebyla.

V rámci Ribbon prvku je však možné přistoupit i k objektům, které nejsou obsaženy v tomto prvků. Jedná se třeba o *Quick Access Toolbar*, standardní (kontextová) menu či *Status Bar*.

Při práci s ovládacím prvkem Ribbon je k dispozici celá škála běžných ovládacích prvků, které je možné na něj umístit. Ať už se jedná o tlačítka pro práci s tabulkami, grafy a různé popisky, nově je k dispozici tzv. *Toggle* tlačítko, které funguje tak, že po jeho stisku se objeví menu. Dále *Split* tlačítko, které vizuálně odděluje prvky v rámci skupiny, zaškrťovací tlačítka, editovací boxy, objekt s názvem *DialogBoxLancher*, což je tlačítko, po jehož stisknutí se objeví dialog box, *DropDown* objekty, *Combo Box* a zvláštní objekt *Galerie*.

Pokud se srovná počet a možnosti ovládacích prvků, které jsou obsaženy v Office 2003 a Office 2007, je zřejmé, že Office 2007 přináší spoustu nových a užitečných ovládacích prvků, neboť ve starší verzi je k dispozici pouze tlačítko, editovací box, *Popup* okno, *Combo Box* a *DropDown Box*.

6.2 XML schéma a XML formát

Jak je již zmíněno výše, změny v Ribbon prvku je možné provádět pomocí XML dokumentu a proto je nutné znát schéma těchto dokumentů. V rámci dokumentu se musí pomocí elementů určit, uprostřed kterého kontejneru se budou změny provádět. K dispozici jsou elementy, vlastnosti a *CallBack* funkce. Pro zobrazení samotného Ribbon prvku je zapotřebí zkopírovat automaticky vygenerovaný kód do třídy *ThisAddin*, *ThisWorkbook*, nebo *ThisDocument*. Kód vypadá např. následovně (název *MujRibbonXML* je konkrétní název vlastního Ribbon prvku).

```
Protected Overrides Function CreateRibbonExtensibilityObject() _  
    As Microsoft.Office.Core.IRibbonExtensibility  
    Return New MujRibbonXML()  
End Function
```

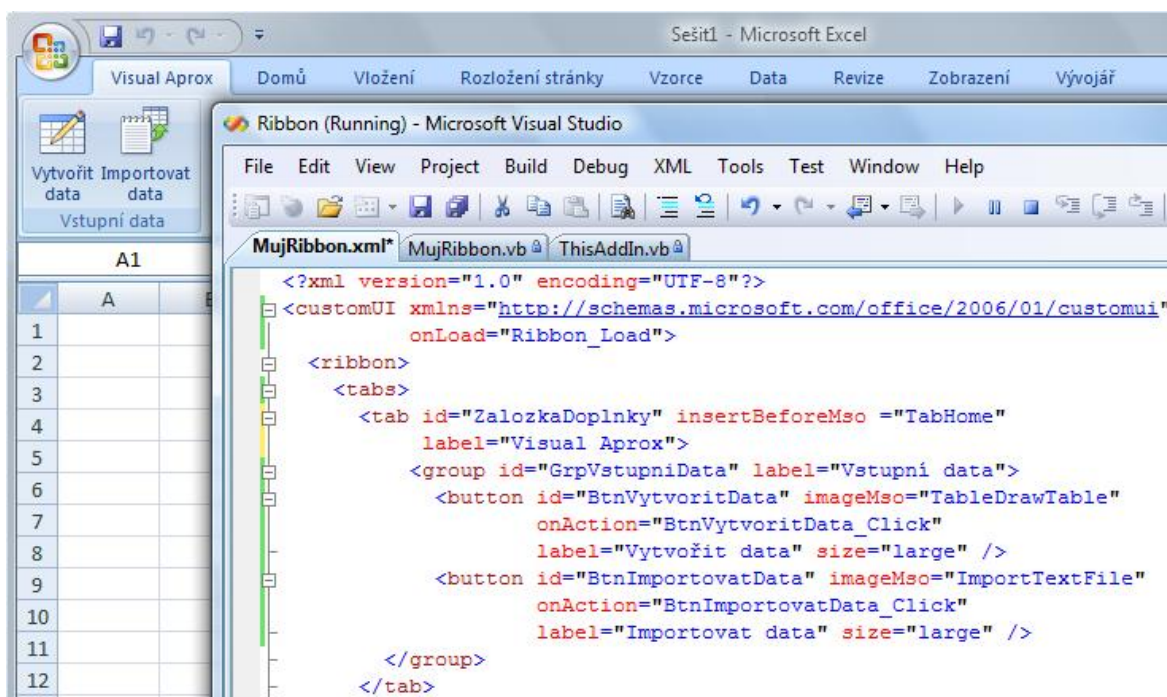
Obr. 19 Kód nutný pro zobrazení Ribbon prvku

Při práci se využívají **elementy** (*Ribbon*, *Tab*, *Group*, *Box*), **vlastnosti** (*id*, *idMso*, *insertAfterMso*, *insertBeforeMso*, *image*, *imageMso*) a **CallBack funkce** (např. *getImage*, *onAction*, *getEnabled*, *getVisible*, *getPressed* apod.). Funguje tzv. model „najdi nebo vytvoř“, což znamená, že určený prvek se buď vytvoří (pokud neexistuje), nebo se pracuje s již existujícím prvkem.

Je možné vytvořit a pojmenovat vlastní prvek, pomocí vlastnosti *id*, nebo určit již existující standardní objekt, který je součástí Office 2007, pomocí vlastnosti *idMso*. V případě práce se standardními, existujícími prvky jsou k dispozici vlastnosti *insertAfterMso* a *insertBeforeMso*. Umožňují určit to, zda bude vlastní vytvořený prvek umístěn před nebo za standardním, existujícím objektem. Lze určit také obrázek pomocí

vlastnosti *image*, nebo se odkázat na již existující obrázek, který je součástí Office, pomocí vlastnosti *imageMso*. Další důležité *CallBack* funkce, které se velmi často používají, jsou *getImage* (pro získání obrázku), *onAction* (co se stane, když nastane událost), *getEnabled*, *getVisible* a *getPressed*. **Všechny operace v rámci Ribbon prvku probíhají na základě volání *CallBack* funkcí.** Veškeré *CallBack* funkce a Ribbon atributy je možno najít v [RICE, F., GETZ, K. 2008].

Ukázka vlastního Ribbon prvku v aplikaci Microsoft Excel a jeho XML dokument je vidět na obr. 20.



Obr. 20 Ribbon prvek a jeho XML dokument

Tento příklad ukazuje, jak rozšířit Ribbon prvek o novou (vlastní) záložku s názvem *Visual Aprox*, která obsahuje skupinu s názvem *Vstupní data*. V rámci této skupiny je pak tlačítko s popiskem *Vytvořit data* a další tlačítko *Importovat data*. Těmto tlačítkům jsou přiřazeny *CallBack* funkce `onAction="BtnVytvoritData_Click"` a `onAction="BtnImportovatData_Click"`. Tyto funkce se volají následujícím způsobem.


```
#Region "Ribbon Callbacks"
'Create callback methods here. For more information about adding callback:
Public Sub Ribbon_Load(ByVal ribbonUI As Office.IRibbonUI)
    Me.ribbon = ribbonUI
End Sub

Public Sub BtnVytvoritData_Click(ByVal control As Office.IRibbonControl)
    'Tělo funkce
End Sub

Public Sub BtnImportovatData_Click(ByVal control As Office.IRibbonControl)
    'Tělo funkce
End Sub
#End Region
```

Obr. 21 Volání CallBack funkcí

To, že se vlastní záložka objevila jako první v seznamu záložek má za následek vlastnost záložky `insertBeforeMso="TabHome"` a to, že mají tlačítka obrázky ikon z prostředí Office 2007 je způsobeno vlastností `imageMso="TableDrawTable"` u prvního tlačítka, případně `imageMso="ImportTextFile"` u tlačítka druhého.

6.3 Dynamická aktualizace

Dynamická aktualizace Ribbon prvku (tlačítka se objevují, mizí apod.) se provádí jen pomocí XML dokumentu. Neexistují žádné API funkce pro operace vložení či odebrání nějakého prvku. Vše je přístupné pouze přes *CallBack* funkce jako jsou např. *setVisible*, *getLabel*, *getImage* apod. [NEUWIRTH, M. 2006].

Funguje to tedy tak, že se na začátku musí vygenerovat všechny potřebné objekty do Ribbon prvku a přes již zmíněné *CallBack* funkce se při počáteční inicializaci musí odpovídající prvky schovat.

Je však možné přidávat dynamicky položky do objektu Galerie a Menu. Dynamická aktualizace se provádí voláním dvou metod a to pomocí metody *Invalidate()*, která aktualizuje celý Ribbon prvek, nebo pomocí metody *InvalidateControl()*, která aktualizuje pouze určitý ovládací prvek [RICE, F., GETZ, K. 2006 A].

XML:

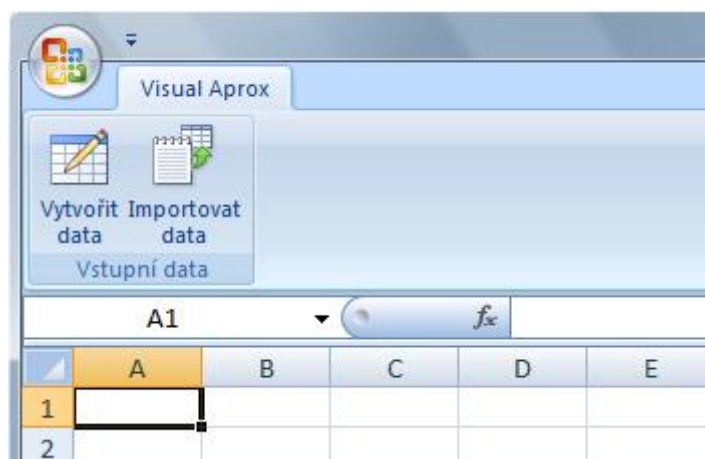
```
<button id="BtnImportovatData" imageMso="ImportTextFile" />
```

VB.NET:

```
Me.ribbon.InvalidateControl("BtnImportovatData")
```

Zajímavou vlastností celého Ribbon prvku je možnost skrýt všechny standardní záložky, jež jsou obsaženy v Office 2007. Tato vlastnost se nazývá *startFromScratch* a je

potřeba ji nastavit jako *True* v případě, že programátor vyžaduje spuštění aplikace pouze se svou vlastní záložkou. Více o vlastnosti *StartFromScratch* je v [RICE, F., GETZ, K. 2006 B].



Obr. 22 Vlastnost *startFromScratch* nastavena na *true*

Jak je vidět na obr. 22, tato vlastnost neovlivňuje pouze záložky, ale zmizí vše i z *Quick Access Toolbaru* a v nabídce *Office Menu* zbudou k dispozici pouze položky *Nový*, *Otevřít*, *Uložit*, *Možnosti* a *Ukončit*.

6.4 Projekty pro Ribbon prvky

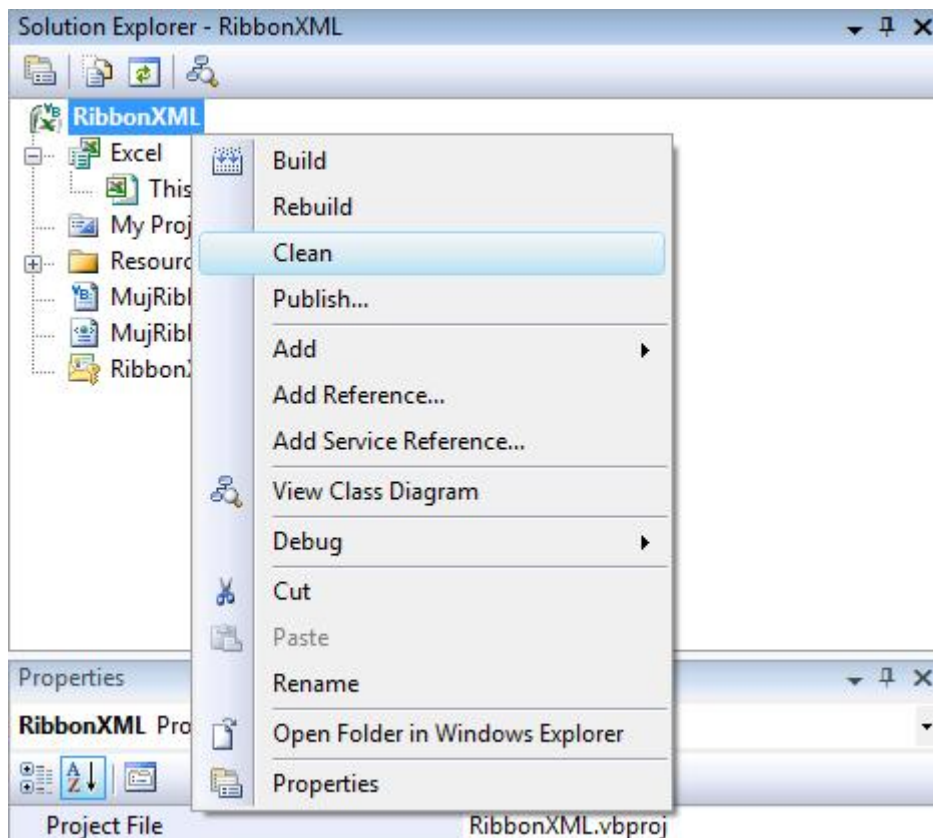
Při vytváření projektů je možné používat tři programové přístupy. COM, VBA a VSTO a to jak pro aplikaci, tak pro dokumenty. COM technologií se vytváří COM doplňky, které se instalují v obou případech na úrovni aplikace. Při řešení pro aplikaci je tento doplněk vidět pořád, v rámci celé aplikace (pro všechny dokumenty) a při řešení pro jednotlivé dokumenty se při přecházení mezi těmito dokumenty Ribbon prvek objevuje nebo mizí. V případě přístupu přes VBA se u řešení pro aplikaci využívá VBA na úrovni globálních šablon pro Word, doplněk pro Excel (*.xlam), nebo doplněk pro PowerPoint (*.ppam). U řešení pro jednotlivé dokumenty se kód vkládá přímo do dokumentu programu Word, Excel nebo PowerPoint. Je tedy jeho součástí a s ním se i přenáší. Kód je možné vkládat i do Access databází. V rámci VSTO lze vytvořit řešení pro celou aplikaci, nebo pro jednotlivé dokumenty.

Tab. 5 Programové přístupy pro vytváření projektů

	COM	VBA	VSTO
Pro aplikaci	Instalace na úrovni aplikace, je pořád vidět	Word Global Template, Excel Add-in (*.xlam), PPT Add-in (*.ppam)	Vytvoření VSTO řešení pro celou aplikaci
Pro dokument	Instalace na úrovni aplikace, zobrazuje se dle zobrazených dokumentů	Word, Excel, PPT dokumenty, Access databáze	Vytvoření VSTO řešení pro dokumenty

6.5 Tipy pro práci s Ribbon prvkem

Po spuštění vlastního řešení z prostředí Visual Studia a následného ukončení je nutné projekt tzv. „vyčistit“. Provádí se to v *Solution Exploreru* pravým kliknutím myši na projekt a volbou *Clean* (viz obr. 23).



Obr. 23 Solution Explorer a „vyčištění“ projektu

V případě nevyčištění projektu vše zůstává uloženo v registrech a i přes ukončení práce s projektem ve Visual Studiu zůstává vlastní Ribbon prvek implementován v celém prostředí dané aplikace balíku Office (pokud je tvořen pomocí doplňku na úrovni aplikace).

V případě vyčištění projektu se rozpracované řešení plně odebere z registrů a z celého Office prostředí, takže se nemůže stát, že si uživatel spustí na počítači jednu z aplikací sady Office a objeví se mu nová záložka s rozpracovaným řešením.

Pokud chce vývojář pracovat s Ribbon prvkem pomocí XML dokumentů a vkládat své vlastní skupiny mezi již existující skupiny obsažené standardně v Ribbon prvku Office 2007, nebo využívat standardních ikon použitých v Office 2007, je nutné, aby znal jejich přesné názvy a to:

- **idMso** – Pro zjištění názvů záložek, kontextových menu, skupin a jednotlivých ovládacích prvků slouží dokument Excel: „2007 Office System Document: Lists of Control IDs“ dostupný z:
<http://www.microsoft.com/downloads/details.aspx?familyid=4329d9e9-4d11-46a5-898d-23e4f331e9ae&displaylang=en>
- **imageMso** – Pro zjištění názvu ikony slouží dokument Excel: „2007 Office System Add-In: Icons Gallery“ dostupný z:
<http://www.microsoft.com/downloads/details.aspx?familyid=12B99325-93E8-4ED4-8385-74D0F7661318&displaylang=en>

Jak postupovat při tvorbě Ribbon prvku?

- Začněte tvořit nejlépe v design módu. Je to rychlejší a přehlednější způsob práce s Ribbon prvkem, než pomocí XML dokumentu.
- V případě nutnosti převed'te Ribbon prvek z design módu na XML dokument – provádí se jedním kliknutím na Ribbon prvku (šipka vpravo nahoře).
- Ponechejte uživatelské rozhraní relativně neměnné. Popisky, ikony a viditelnost jednotlivých ovládacích prvků většinou za běhu není dobré měnit.
- Podokna úloh a dialogová okna by se neměla v dokumentu objevovat automaticky, ale pouze po volání uživatelské akce (např. stisknutí tlačítka). Ideálně jsou všechna podokna úloh otevírána použitím tlačítka na Ribbon prvku.
- Logicky a chronologicky seřaďte ovládací prvky na vlastních záložkách. Tzn. shora dolů, zleva doprava, podle předpokládaného využívání ovládacích prvků a postupu práce s nimi. Neduplikujte příkazy.

Na co si dát pozor (z pohledu minimalizace potíží s jiným řešením)?

- Neskrývejte vestavěné záložky nebo skupiny. Jiné doplňky mohou do těchto záložek přidávat ovládací prvky.
- Nepoužívejte běžně vlastnost `startFromScratch`, jen pokud je dané přizpůsobení připojeno ke konkrétnímu dokumentu. Vlastnost `startFromScratch` skryje celé vestavěné uživatelské rozhraní a je užitečná pouze pro vytváření kompletně přizpůsobených, účelně stavěných řešení.
- Nepoužívejte příkaz, který zakáže nebo změní chování vestavěných příkazů anebo změni vestavěný příkaz jakýmkoli jiným způsobem. Jiná řešení mohou být závislá na určitém chování těchto vestavěných příkazů.

Jak optimálně rozšířit vlastní řešení a umístit položky v Ribbon prvku?

- Rozšiřte Ribbon prvek, pokud dané řešení obsahuje příkazy, které ovlivňují obsah dokumentu.
- Rozšiřte Microsoft Office menu (*File Menu*), jestliže dané řešení přidává příkazy, které ovlivňují dokument jako celek.
- Rozšiřte sadu kontextuálních záložek, v případě, že dané řešení obsahuje příkazy, které obsluhují jeden z vestavěných Office objektových typů jako jsou obrázky, tabulky a grafy. Přidejte svou vlastní záložku do sady kontextuálních záložek, pokud máte dostatek příkazů pro vyplnění záložky. Jestliže nemáte, přidejte příkaz jako skupinu v existující kontextuální záložce.
- Vytvořte vlastní podokno úloh, pokud dané řešení potřebuje zobrazovat data o dokumentu (je vyžadováno, aby byl dokument viditelný). Zobrazujte toto podokno úloh pouze na základě uživatelských událostí (např. stisknutí tlačítka).
- V případě potřeby použijte vlastní ovládací prvky, které jsou mimo Ribbon prvek.

Optimální umístění položek v Ribbon prvku je zobrazeno také v tab. 6. Více je možné dozvědět se např. v [MICROSOFT. 2006 B].

Tab. 6 Optimální umístění položek v Ribbon prvku

UI	Obsahuje
File Menu	Externí příkazy nesouvisející s dokumentem
Záložky	Příkazy pro modifikaci obsahu dokumentu
Kontextuální záložky	Specifické příkazy pro jednotlivé objekty v dokumentech
Vlastní podokna úloh	Informace relevantní aktuálnímu dokumentu

7. Dokumentace k doplňku Visual Aprox

Vytvořený doplněk Visual Aprox je součástí nového uživatelského rozhraní aplikace Microsoft Excel 2007. Po spuštění aplikace se doplněk zavede a zobrazí se jako nová záložka s vlastními ovládacími prvky na konci Ribbon panelu (neboli pásu karet). Tento doplněk se instaluje jako klasická aplikace. Vytvořený instalační balíček lze přehrát na pevný disk např. z USB disku, nebo jej lze stáhnout z webového serveru. Na obr. 24 je vidět celá vytvořená záložka s vlastními skupinami ovládacích prvků.



Obr. 24 Nová záložka Visual Aprox

První skupina *Vstupní data* slouží pro vytváření vlastních dat, importování experimentálních dat a generování dat přechodových charakteristik. V této skupině je také tlačítko pro odstranění dat z aktivního listu. Další skupinou jsou *Grafy*, kde je tlačítko pro vykreslení grafu dat, exportování grafu do prostředí Word 2007 a také pro odstranění řady z grafu. Další, v pořadí třetí skupinou je *Experimentální identifikace*, kde se za pomoci příslušného tlačítka vypočítá přechodová charakteristika dle zadaných parametrů a vykreslí do grafu. Následující skupinou jsou *Aproximace*. Tlačítka v této skupině umožňují provádět aproximace přímkou, polynome 2. až 6. stupně a exponenciálou. V poslední skupině nazvané *Ostatní* je možno přejít na některé webové stránky, zapsat si poznámky, spustit nápovědu nebo zjistit informace o doplňku.

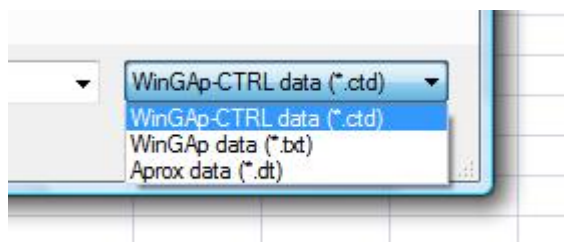
7.1 Popis nového doplňku

Nově vyvinuté funkce v doplňku rozšiřují možnosti nástroje Microsoft Excel pro konkrétní účel zpracování a prezentace zadaných či experimentálně získaných dat. Ribbon prvek je rozdělen do tematických skupin, které budou nyní popsány.

Skupina Vstupní data:**Obr. 25 Vzhled skupiny Vstupní data**

V této skupině ovládacích prvků jsou tlačítka, jež umožňují vytvářet vlastní data, importovat experimentálně získaná data a také generovat, nebo odstranit data. Vytváření vlastních dat se provádí přímým zadáváním hodnot do buněk sešitu MS Excel. K tomu příslušné tlačítko tedy slouží pro naformátování tabulky a vyplnění záhlaví. Pro zaktivnění tlačítka pro vykreslení grafu je nutné použít tlačítko vytvořit data, nebo importovat data.

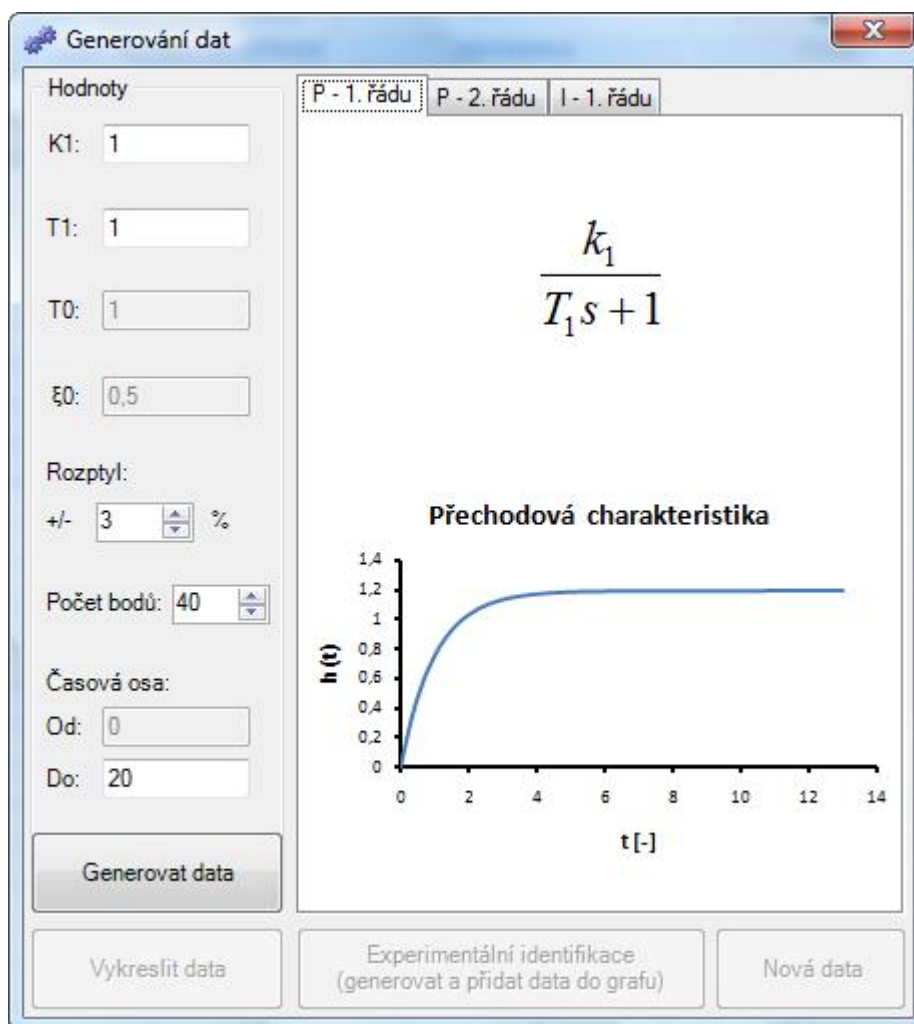
Po stisknutí tlačítka importovat data se otevře *Open File Dialog*, který podporuje načítání formátu souborů s příponou *.ctd, *.txt a *.dt (viz obr. 26). Soubory *.ctd jsou importovány ze současně používaného programového modulu sloužícího převážně pro komunikaci s teplovzdušným modelem a pro záznam dat. Tento teplovzdušný model slouží pro výuku v laboratořích na katedře Automatizační techniky a řízení.

**Obr. 26 Podporované formáty souboru**

Soubory *.txt a *.dt jsou používány v předchozích programových modulech a v rámci zajištění určité kompatibility s těmito staršími formáty souborů je v doplňku poskytnuta podpora pro načítání těchto formátů souborů. Navíc současně používaný program pro měření experimentálních dat umožňuje export svých dat z formátu *.ctd do formátu předchozího programového modulu pro zpracování a prezentaci dat *.txt. Formát *.dt lze jen stěží někde najít, neboť jde o formát souboru z velmi starého programového modulu, který je spustitelný pouze z prostředí MS-DOS či z příkazové řádky Windows XP a dnes se již vůbec nepoužívá.

Tlačítko generování dat slouží pro výpočet a vykreslení přechodových charakteristik proporcionálního členu se setrvačností 1. řádu, proporcionálního členu se setrvačností 2. řádu a integračního členu se setrvačností 1. řádu. U přenosu proporcionálního členu se setrvačností 2. řádu je omezení bezrozměrného koeficientu tlumení $0 \leq \xi \leq 1$. V budoucnu

by bylo vhodné toto omezení změnit na $\xi \geq 0$, což by zahrnovalo i členy bez překmitu s vyšší hodnotou ξ , která stále ovlivňuje průběh přechodové charakteristiky. Tato generovaná data lze ihned vykreslovat do grafu a poté, pomocí zadávání parametrů v příslušném formuláři experimentálně zjišťovat parametry soustavy (viz obr. 27).



Obr. 27 Formulář pro generování dat

Samotné generování počítá s určitým rozptylem, který je implicitně nastaven na hodnotu $\pm 3\%$. Tento rozptyl je možné měnit v rozmezí $0 \div 5\%$ a má simulovat experimentálně změřená data. Výsledné hodnoty jsou tímto rozptylem násobeny za pomoci funkce *Random*, aby bylo zajištěno, že i při všech stejných parametrech bude výstup vždy jiný. Po vykreslení generovaných dat je možno provádět experimentální identifikaci, kde již hodnota rozptylu nehraje roli, protože ve výpočtu není zahrnuta. Získá se tak přesný průběh podle uživatelem určených parametrů soustavy.

Jako poslední je ve skupině Vstupní data umístěno tlačítko pro vymazání dat, pomocí kterého může uživatel veškerá data rychle a jednoduše vymazat.

Skupina Grafy:**Obr. 28 Vzhled skupiny Grafy**

Tato skupina ovládacích prvků obsahuje tlačítka pro vykreslení a exportování grafu a také pro odstranění řady z grafu. Pro zaktivnění tlačítka pro vykreslení grafu je nutné použít tlačítko vytvořit data, nebo importovat data. Pro samotné vykreslení grafu je uživateli nabídnuto ze dvou možností, a to vykreslit graf všech dat (ať už vytvořených či importovaných) nebo graf vybraných dat. U volby grafu vybraných dat se otevře *inputbox*, kde je možno zadat rozsah buněk ručně, nebo vybrat danou oblast v listu.

Pro možnost použití grafů pro vypracování protokolů z měření byla sestavena funkce pro automatické exportování grafu do prostředí Word 2007. Tento export není tvořen zkopírováním obrazovky, ale objektu (grafu) a vložením do uvedeného prostředí. Tím pádem má uživatel možnost graf editovat i v prostředí Word a to včetně zdrojových dat apod.

U vykreslených grafů je možno pomocí příslušného tlačítka odstraňovat jednotlivé řady, ze kterých je graf sestaven. Tuto funkci poskytuje i samotný Excel, ale ten odstraňuje pouze řadu z grafu, nikoli i její zdrojová data. Ve vytvořeném řešení se odstraní rovněž data, z důvodu možného velkého množství vykreslených řad v grafu, což může nastat, pokud bude uživatel dlouho experimentálně hledat parametry soustavy.

Skupina Experimentální identifikace:**Obr. 29 Vzhled skupiny Experimentální identifikace**

Jak již bylo zmíněno výše, uživatel si může vyzkoušet odhad parametrů soustavy cvičně, pomocí formuláře pro generování dat. Tato skupina ovládacích prvků, umístěna přímo na Ribbon prvku, však slouží pro vkládání průběhů přechodových charakteristik přímo do experimentálně získaných a vykreslených dat, které byly předem importovány.

Nejprve je potřeba zvolit člen, který se bude počítat, následně zvolit parametry soustavy a spustit výpočet, který po dokončení vypočtená data ihned vykreslí do grafu. Při

volbě členu se automaticky podle vybraného členu povolují či zakazují příslušná políčka s parametry.

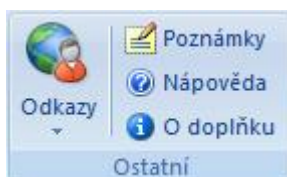
Skupina *Aproximace*:



Obr. 30 Vzhled skupiny *Aproximace*

V této skupině ovládacích prvků jsou tlačítka, jež umožňují aproximaci dat a to přímkou, polynomem (2. až 6. stupně) a exponenciálou. Aproximace jsou počítány metodou nejmenších čtverců, která minimalizuje součet čtverců odchylek mezi hodnotami aproximující funkce a měřenými hodnotami ve všech bodech. Po výpočtu se zobrazí nová (aproximovaná) funkce v grafu i s příslušnými koeficienty aproximace a hodnotou spolehlivosti. Podle této hodnoty lze usoudit, zda je provedená aproximace vyhovující či nikoli, protože se v ideálním případě blíží tato hodnota co nejvíce číslu jedna.

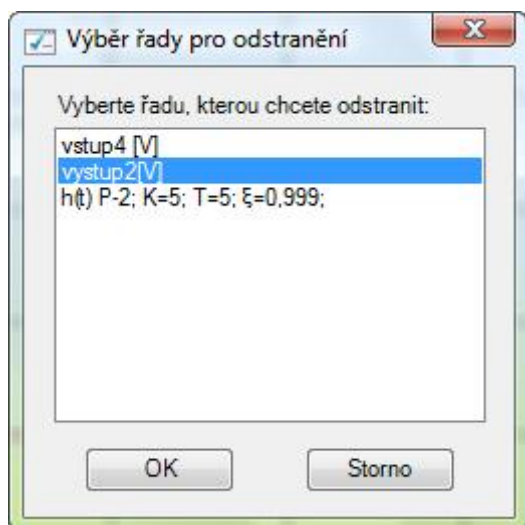
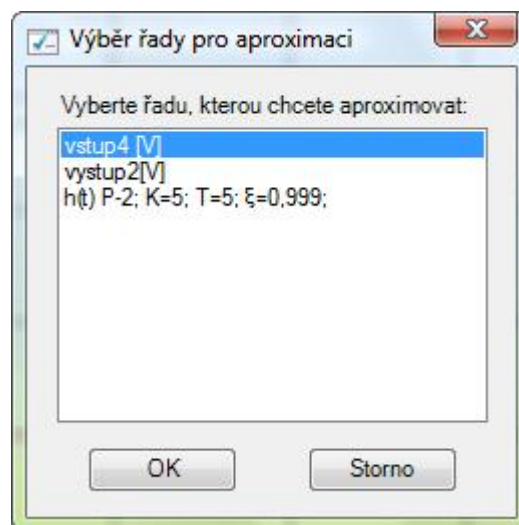
Skupina *Ostatní*:



Obr. 31 Vzhled skupiny *Ostatní*

Poslední skupina ovládacích prvků obsahuje odkaz na webové stránky laboratoře katedry Automatizační techniky a řízení a některé další důležité odkazy. Odkaz na webové stránky laboratoře studenti často využívají, neboť jsou na nich uvedeny návody do cvičení. Následující odkazy patří webovým stránkám katedry Automatizační techniky a řízení, fakulty strojní a Vysoké školy báňské – Technické univerzity Ostrava.

Další tlačítko v této skupině ovládacích prvků umožňuje zapsat si v průběhu práce jakékoli poznámky, které zůstávají ve formuláři uloženy až do doby vypnutí programu Excel. Tyto poznámky lze pak jednoduše zkopírovat např. do prostředí Word. V případě potíží obsahuje poslední skupina i tlačítko pro nápovědu v podobě HTML stránek, kde je možné najít dokumentaci k doplňku Visual Aprox, jeho systémové požadavky, popis uživatelského prostředí apod. Nakonec je zde i tlačítko pro zobrazení informací o doplňku, kde je uveden název, aktuálně používaná verze apod.

Další formuláře obsažené v doplňku:**Obr. 32 Formulář pro odstranění řady****Obr. 33 Formulář pro aproximaci řady**

V doplňku je použit ještě jeden formulář, který má dvě podoby a používá se pro odstranění řady nebo aproximaci určité řady. Tento formulář se objeví po stisku tlačítka odstranit řadu, nebo po stisku jednoho z tlačítek určených pro aproximaci. Do formuláře se programově načte záhlaví tabulky z listu Data a změní se název formuláře, popisek a jeho celková funkčnost v závislosti na tom, kterým tlačítkem byl formulář zavolán. Tlačítko OK se zaktivní až po výběru řady. Potvrzení vybrané řady lze provést též dvojklikem na danou řadu.

7.2 Požadavky, vlastnosti a možnosti vytvořeného doplňku

Systémové požadavky:

Na tuto diplomovou práci, tedy i vytvořený doplněk se plně vztahuje zákon č. 121/2000 Sb. – autorský zákon. Vysoká škola báňská – Technická univerzita Ostrava má právo nevýdělečně ke své vnitřní potřebě tuto diplomovou práci, tedy i vytvořený doplněk užít (§35 odst. 3).

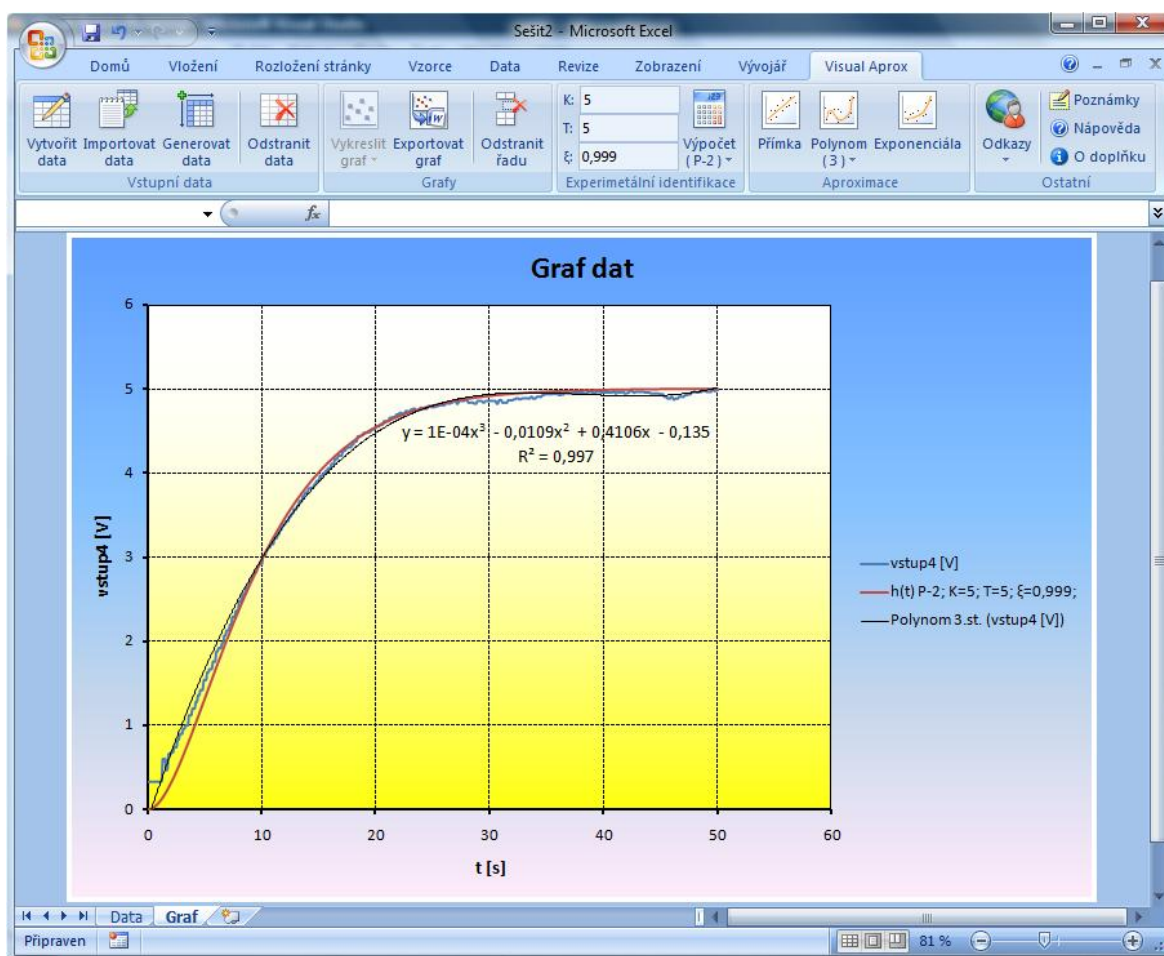
Nároky na hardware jsou určeny pouze operačním systémem, pod kterým aplikace pracuje. Co se týče softwaru, tak pro plnou funkčnost doplňku je vyžadován MS Excel 2007, MS Word 2007 a .NET Framework 3.5. Doplněk je vyvíjen v jazyce Visual Basic .NET a jelikož je na aplikační úrovni programu Excel, je potřeba jej instalovat. To lze provést buďto lokálně např. z USB disku či CD, nebo s využitím webového serveru. Při práci nemusí být uživatel připojen k síti internet. Správná funkčnost doplňku nijak nezávisí na nastavení národního prostředí OS Windows.

Uživatelské prostředí:

Uživatelské prostředí není třeba zdlouhavě popisovat, neboť prostředí programu Excel, nebo obecně programů z balíku Microsoft Office je většině uživatelů dobře známé. Jediné, co se výrazně změnilo ve verzi Office 2007 je nový ovládací panel zvaný Ribbon prvek. To může být pro uživatele částečné znepríjemnění práce, avšak tento dojem přetrvává jen do doby, kdy si na nové prostředí zvykne a zjistí, že je opravdu mnohem přehlednější a pracuje se v něm výrazně rychleji, než v předchozích verzích. Doplněk obsahující vlastní záložku v Ribbon prvku je dobře uspořádaný a jednoduchý na ovládání díky dobře rozvrženým skupinám ovládacích prvků. Při práci jsou data přehledně rozdělena do pomyslné tabulky, skládající se z řádků (označených číslicemi) a sloupců (označených písmeny).

	A	B	C
1	t [s]	vstup4 [V]	h(t) P=2; K=5; T=5; $\xi=0,999$;
2	0	0,33	-4,42897E-16
3	0,1	0,33	0,000986779
4	0,2	0,33	0,003895019
5	0,3	0,33	0,008648311
6	0,4	0,33	0,015172517
7	0,5	0,33	0,023395709
8	0,6	0,33	0,033248109
9	0,7	0,33	0,044662034
10	0,8	0,33	0,057571842
11	0,9	0,33	0,071913872
12	1	0,33	0,087626399
13	1,1	0,33	0,104649576
14	1,2	0,33	0,122925387
15	1,3	0,6	0,142397599

Obr. 34 Pracovní prostředí s daty



Obr. 35 Pracovní prostředí s vykresleným grafem

Doplňek podporuje obsluhu ovládacích prvků pomocí horkých kláves, což umožňuje rychlejší ovládání. Modul je vybaven i nápovědou, kde je uveden popis a ovládání doplňku a systémové požadavky.

Vstupy a výstupy:

Hodnoty se zadávají do sloupců. První z nich je nezávisle proměnná, ostatní jsou závisle proměnné. Doplňek tedy pracuje s jednou nezávisle proměnnou a více závisle proměnnými. Počet zadaných hodnot v jednom sloupci je omezen na maximální počet 32000 hodnot, což pro konkrétní použití nadměrně postačuje. Doplňek poskytuje podporu načítání několika konkrétních formátů souborů, mezi něž patří *.ctd, *.txt a *.dt a program jako celek podporuje načítání dat z textového formátu s různými oddělovači, z databázových souborů apod. Vlastní data i grafy umístěné v pracovním sešitu je možné ukládat v novém datovém formátu *.xlsx. Oproti předchozímu formátu souboru poskytuje tento nový formát ukládání do tzv. Open XML formátu. To znamená, že veškeré informace o datech, grafech, listech, sešitu apod. jsou uloženy do jednotlivých XML dokumentů,

které lze otevřít bez použití nástroje Microsoft Excel. Níže je znázorněn XML dokument náhodně vybraného sešitu. Název XML dokumentu je workbook.xml.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
- <workbook xmlns="http://schemas.openxmlformats.org/spreadsheetml/2006/main"
  xmlns:r="http://schemas.openxmlformats.org/officeDocument/2006/relationships">
  <fileVersion appName="xl" lastEdited="4" lowestEdited="4" rupBuild="4506" />
  <workbookPr defaultThemeVersion="124226" />
- <bookViews>
  <workbookView xWindow="600" yWindow="360" windowWidth="17895" windowHeight="7635" />
</bookViews>
- <sheets>
  <sheet name="Data" sheetId="1" r:id="rId1" />
  <sheet name="Graf" sheetId="4" r:id="rId2" />
  <sheet name="List2" sheetId="2" state="hidden" r:id="rId3" />
  <sheet name="List3" sheetId="3" state="hidden" r:id="rId4" />
</sheets>
  <calcPr calcId="125725" />
</workbook>
```

Obr. 36 Příklad XML zápisu náhodně vybraného sešitu aplikace MS Excel

Veškerá data i grafy je možné libovolně tisknout a nastavovat mnoho vlastností a možností tisku. Grafy je možné také exportovat do prostředí Microsoft Word 2007 z důvodu dalšího použití pro vypracování protokolů z měření apod. Graf není vkládán jako obrázek pomocí *Hard Copy* obrazovky, ale jako objekt. Proto je možné i v prostředí Word tyto grafy, resp. jejich data dodatečně editovat (jsou propojeny s daným sešitem).

Zpracování dat:

Doplňěk umožňuje následující typy výpočtů pro experimentální identifikaci soustavy:

- proporcionální člen se setrvačností 1. řádu
- proporcionální člen se setrvačností 2. řádu s podmínkou $0 \leq \xi \leq 1$
- integrační člen se setrvačností 1. řádu

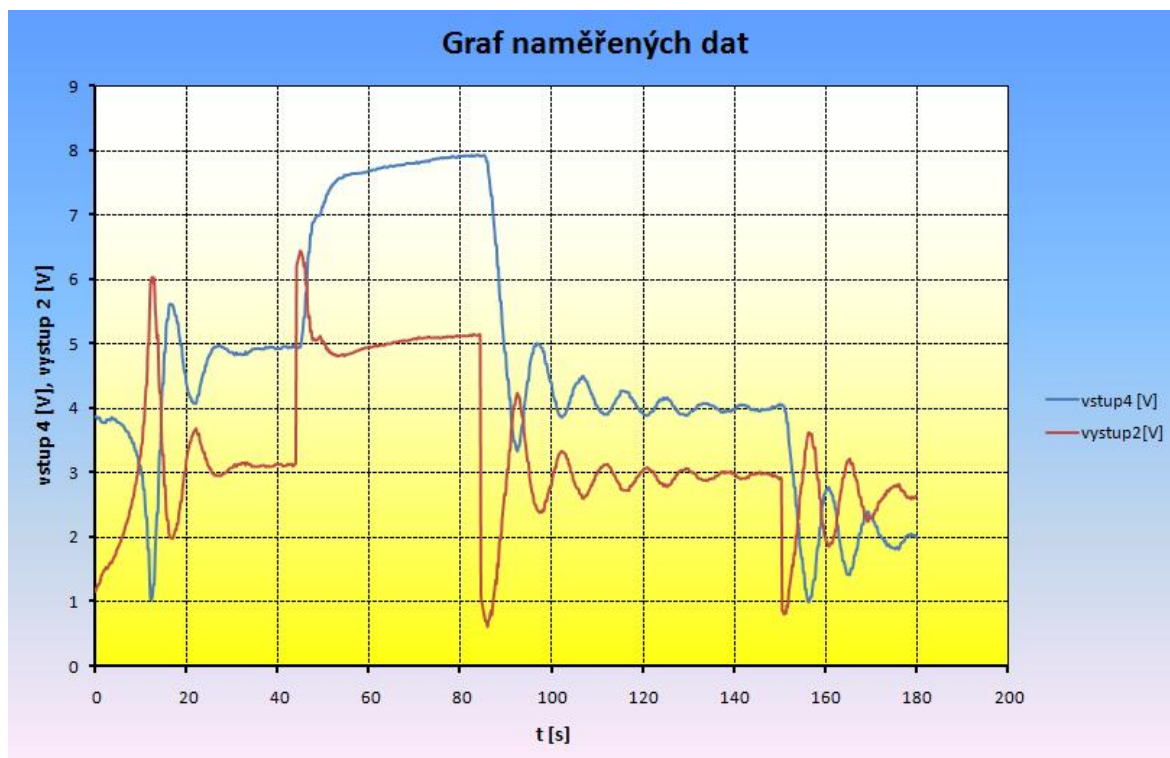
Doplňěk umožňuje následující typy aproximace zadaných či načtených dat:

- přímka: $y = ax + b$
- polynom: $y = a_n x^n + \dots + a_1 x + a_0$ pro $n = 2 \div 6$
- exponenciála: $y = ae^{bx}$

Zobrazení dat:

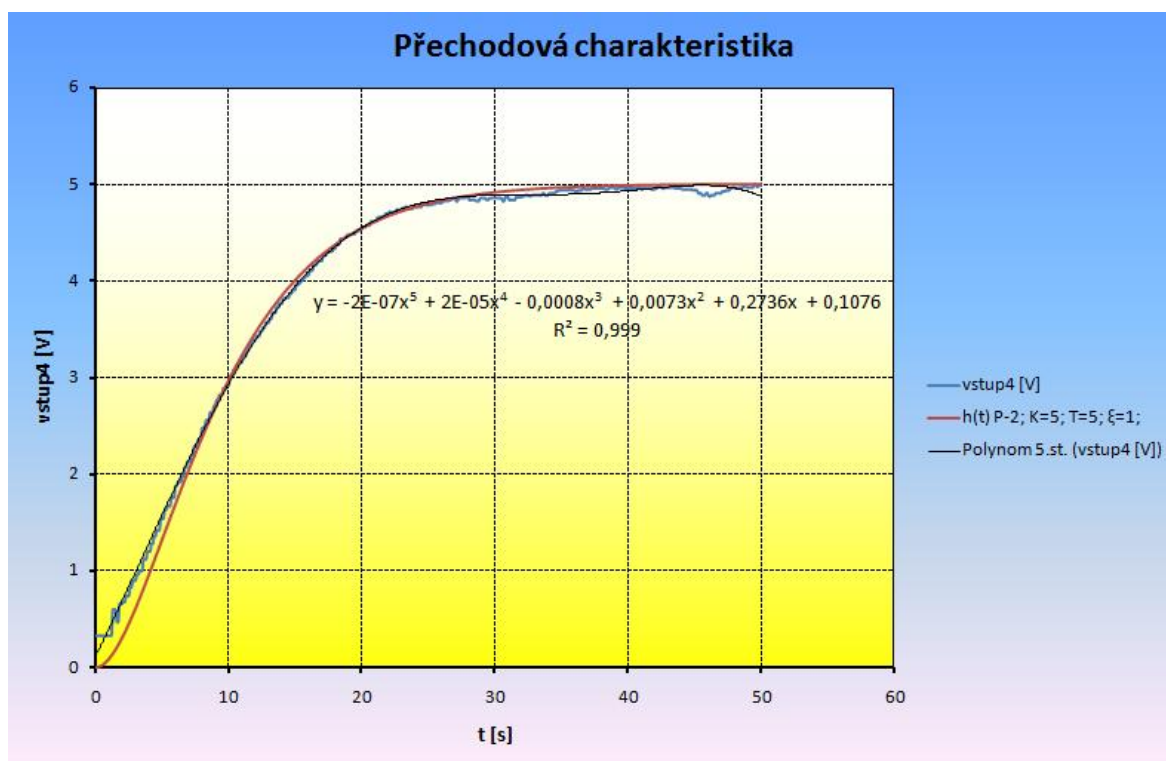
Doplňěk umožňuje vykreslit jeden graf k jednomu souboru dat. V případě nutnosti je možno další grafy vykreslit standardními nástroji programu Excel, což však není předpokládáno. Uživatel má na výběr, zda vykreslí do grafu všechna data, nebo jen vybraná data. V případě zvolení vykreslení grafu vybraných dat se po stisknutí příslušného tlačítka zobrazí *inputbox*, kde je možné zadat rozsah buněk ručně, nebo oblast vybrat

v příslušném listu. Po vykreslení grafu dat je dále umožněno s daty pracovat a to buď počítat data pro zvolený člen (proporcionální člen se setrvačností 1. řádu, proporcionální člen se setrvačností 2. řádu, integrační člen se setrvačností 1. řádu) z důvodu experimentální identifikace soustavy, nebo provádět aproximace, z důvodu nahrazení funkce jinou, jednodušší, matematicky popsanou funkcí. Výsledky těchto výpočtů se v grafech následně ihned zobrazí.



Obr. 37 Graf naměřených dat (1. ukázka)

Na obr. 37 je znázorněn graf experimentálně získaných dat z programu WinGAP-CTRL. Na obr. 38 je pak vidět opět graf experimentálně získaných dat, ale na tento graf byla aplikována experimentální identifikace parametrů soustavy a výsledek je znázorněn v legendě grafu. Z důvodu ukázky byla provedena i aproximace naměřeného průběhu polynomem 5. stupně a v grafu se tak zobrazily koeficienty daného polynomu i s hodnotou spolehlivosti. Na těchto dvou souborech dat zobrazených na obr. 37 a obr. 38 (a také na mnoha dalších) byla ověřena funkčnost vytvořeného programového doplňku. Vše probíhalo standardně, bez jakýchkoli potíží.



Obr. 38 Graf naměřených dat (2. ukázka)

Pro zobrazení dat v grafu existují určité limity, které sice nejsou v tomto případě omezující, ale je vhodné je uvést. Tyto limity jsou:

- Objekt graf se může odkazovat pouze na 255 pracovních sešitů
- Datové řady v grafu nemohou překročit počet 255 řad
- Jednotlivé datové řady nesmí překročit 32000 bodů
- Celkový počet bodů pro všechny datové řady v jednom grafu nesmí překročit počet 256000 bodů

8. Aproximace experimentálních dat

V technické praxi se často získávají informace o probíhajících procesech měřením nějaké fyzikální veličiny. Výsledkem měření je obvykle tabulka experimentálně získaných hodnot, která však není pro další zpracování příliš vhodná. Lépe se pracuje se závislostí, která je vyjádřena analytickým předpisem [FINKEOVÁ, J. A KOL. 2006]. Ten umožňuje např. analyzovat, jakým způsobem ovlivňují hodnoty nezávisle proměnných hodnotu závisle proměnné apod. Proto je dobré experimentálně získaná data aproximovat vhodným typem aproximační funkce a určit hodnoty jejích koeficientů tak, aby průběh funkce co nejlépe vystihoval naměřená data.

Obecná metoda, jak vybrat vhodný typ aproximační funkce, neexistuje. Většinou ale tvar funkce vyplývá z fyzikálních souvislostí v experimentech a je známý. Přesto se někdy musí zvolit typ funkce intuitivně, podle znalostí a zkušeností, charakteru a trendu závislosti, které lze posoudit po vykreslení experimentálně získaných dat do grafu. Data se obvykle prokládají nějakou jednodušší funkcí, např. přímkou, polynomem, exponenciálou, jejich kombinací apod. Nejčastěji používaným typem aproximační funkce je polynom, kam patří přímka či parabola a druhým nejrozšířenějším typem je trigonometrický polynom [FINKEOVÁ, J. A KOL. 2006].

Hodnoty koeficientů aproximační funkce se určují pomocí matematických metod. Tyto metody hledají takové koeficienty aproximační funkce, pro které nabývá účelová funkce minima. Účelových funkcí se používá několik, a to na základě předpokladů o chybách měření, kterých je možné se dopustit. Známé jsou např. tyto:

- Metoda nejmenších čtverců
- Metodu maximální věrohodnosti
- Čebyševova metoda (metoda „minimaxu“)
- Metoda součtu absolutních chyb

V uvedených metodách není napsána aproximace dat interpolací. Ta totiž prokládá data tak, aby byly hodnoty v uzlových bodech měření stejné jako aproximující hodnoty. Vzhledem k tomu, že experimentálně naměřená data jsou vždy zatížena chybou měření, nemělo by účel tyto hodnoty přesně respektovat. Z tohoto důvodu není vhodné k aproximaci vztahu mezi veličinami metodu interpolace použít. Proto se také v těchto případech hledá nějaká jednoduchá funkce, která respektuje pouze charakter závislosti veličin, tak, aby celková chyba aproximace byla co nejmenší.

Je dokázáno, že za splnění předpokladů o chybách měření se spolehlivých výsledků dosahuje metodou nejmenších čtverců. Tato metoda je v současné době nejpoužívanější metodou pro aproximaci experimentálně získaných dat. Metoda sice vyžaduje provedení velkého počtu aritmetických operací, ale v dnešní době vyspělé výpočetní techniky to není problém. Lze s ní efektivně zpracovat i rozsáhlé soubory experimentálních dat. Více je možno dozvědět se např. v [FINKEOVÁ, J. A KOL. 2006, HORÁLEK, J.].

V doplňku je použita metoda nejmenších čtverců, která minimalizuje součet čtverců odchylek mezi hodnotami aproximující funkce a měřenými hodnotami ve všech uzlových bodech. Je umožněno provádět aproximace přímkou, polynomem (2. až 6. stupně) a exponenciálou. O metodě nejmenších čtverců je možno dozvědět se více např. z [WIKIPEDIE. 2009].

9. Závěr

Práce obsahuje přehled všech možných požadavků na programový modul obecně. Byly nastíněny systémové požadavky programů, nároky na vstupy a výstupy, uživatelské prostředí a na zpracování a prezentaci samotných dat. V další části byly popsány důležité vlastnosti, kterými by měl programový modul disponovat. Je uveden také konkrétní rozbor jednotlivých programových modulů, realizovaných na katedře Automatizační techniky a řízení, VŠB-TU Ostrava.

Následující kapitola obsahuje informace o architektuře .NET a je doplněna příslušnými obrázky. Tyto informace jsou uvedeny proto, že všechny tři následně uvedené návrhy programového řešení se o .NET technologii přímo opírají. Po zvážení všech výhod a nevýhod jednotlivých řešení, sepsaných v poslední podkapitole, je zvolen doplněk pro aplikaci Microsoft Excel na aplikační úrovni. Tento doplněk je vyvíjen v prostředí *Visual Studio 2008* za pomoci nástroje *Visual Studio Tools for Office*. Proto je tomuto tématu v krátkosti také věnována pozornost. Následně je uvedena architektura VSTO 2008 a naznačena možnost tvorby Ribbon panelu pomocí VBA kódu, kterým lze volat kód napsaný v .NETu. To patří mezi jednu z novinek VSTO 2008.

Další část se zaměřuje na samotný Ribbon prvek, což je nové uživatelské rozhraní v Office 2007. Tento prvek je v práci popsán a je zde vytvořen návod, jak postupovat při tvorbě Ribbonu pomocí XML dokumentu. Je také uvedeno na co si dát pozor a jak optimálně rozšířit vlastní řešení a umístit položky v Ribbon panelu. Nové uživatelské rozhraní lze upravovat pomocí VSTO ve verzi 2008 i pomocí tzv. Ribbon Designeru, ale to není třeba popisovat, neboť se tvorba Ribbon prvku tímto způsobem velice podobá vytváření klasických Windows aplikací, což většina vývojářů zná.

V předposlední kapitole je uvedena samotná dokumentace k nově vytvořenému doplňku a také jeho popis, požadavky, vlastnosti a možnosti. Vše je popsáno a rozděleno podle jednotlivých skupin ovládacích prvků, jež jsou na Ribbon panelu umístěny. Zmínka je taky o formulářích, se kterými se může uživatel v průběhu práce s tímto doplňkem setkat. Ať už jde o složitý formulář pro generování dat nebo formuláře pro odstranění či aproximaci řady, všechny jsou podrobně popsány, aby uživatel dobře věděl, jak s nimi pracovat.

Kapitola aproximace experimentálních dat obsahuje základní informace o aproximacích, proč se používají a k čemu jsou užitečné a jsou zde uvedené aproximace, které byly použity v doplňku.

Vytvořený programový modul byl testován s několika formáty souborů reálných experimentálních dat a projevoval známky maximální stability, funkčnosti, jednoduchosti a uživatelské přívětivosti.

V případě další práce na tomto doplňku by bylo dobré uzpůsobit omezení bezrozměrného koeficientu tlumení u proporcionálního členu se setrvačností 2. řádu na $\xi \geq 0$. Dále možná výpočet parametrů soustavy z dat přechodové charakteristiky, rozšířit možnosti aproximací a také umožnit vykreslení grafu odchylek.

Na závěr své diplomové práce bych rád poděkoval panu prof. Dr. RNDr. Lubomíru Smutnému a panu Ing. Davidu Fojtíkovi, Ph.D. za jejich ochotu a obětavou pomoc při tvorbě této diplomové práce.

Použitá literatura

ANDERSON, T. *Pro Office 2007 Development with VSTO*. Berkeley: Apress, 2009. 285s. ISBN 978-1-4302-1071-9.

BRODA, R. *Programová podpora zpracování experimentálních dat*. Ostrava: 2005. 64s. Diplomová práce VŠB-TU Ostrava.

BRUNEY, A. *Professional VSTO 2005: Visual Studio 2005 Tools for Office*. Indianapolis: Wiley Publishing, 2006. 339s. ISBN 978-0-471-78813-3.

BRŮHA, L. *Začínáme programovat v jazyce Visual Basic .NET*. 1. vyd. Praha: Computer Press, 2002. 302s. ISBN 80-7226-785-X.

BUCHANAN, L., J., TURNER, R., P. *Numerical Methods and Analysis*. Singapore: McGraw-Hill, 1992. 751s. ISBN 0-07-112922-7.

CARTER, E., LIPPERT, E. *Visual Studio Tools for Office: Using Visual Basic 2005 with Excel, Word, Outlook, and InfoPath*. 1st. edition. Massachusetts: Addison Wesley Professional, 2006. 1120s. ISBN 978-0-321-41175-4.

ETHERIDGE, D. *Microsoft Office Excel 2007 Programming*. Indianapolis: Wiley Publishing, 2007. 355s. ISBN 978-0-470-13230-2.

FARANA, R., SMUTNÝ, L., VÍTEČEK, A., VÍTEČKOVÁ, M. *Zpracování závěrečných prací z oblasti automatizace a informatiky*. 1. vyd. Ostrava: VŠB-TU Ostrava, 2004. 116s. ISBN 80-248-0557-X.

FINKEOVÁ, J., NACHTIGALOVÁ, I., POŽIVIL, J. A KOL. 2006. *Aplikace výpočetní techniky*, [online]. HTML format. Dostupné z: <<http://uprt.vscht.cz/ucebnice/AVT/>>

GETZ, K. 2005. *Understanding the Excel Object Model from a Visual Studio 2005 Developer's Perspective*, [online]. HTML formát. MCW Technologies, LLC, 2005 Dostupné z: <[http://msdn.microsoft.com/en-us/library/aa537184\(office.11\).aspx](http://msdn.microsoft.com/en-us/library/aa537184(office.11).aspx)>

HANÁK, J. 2007. *Visual Studio 2008 a Microsoft .NET Framework 3.5*, [online]. HTML formát. ISSN 1212-8554. Dostupné z: <<http://www.zive.cz/Clanky/Visual-Studio-2008-Vyvojsarske-nekonecno/sc-3-a-138579/default.aspx>>

HORÁLEK, J. *Zpracování fyzikálních měření na PC*, [online]. PDF formát. Dostupné z: <<http://www.horalek.org/pdf/text.pdf>>

KAČMÁŘ, D. 2001. *Architektura .NET*, [online]. PDF formát. Dostupné z:

<<http://www.microsoft.com/cze/msdn/csebooks/default.aspx>>

KOPECKÝ, M. *Programová podpora měřicích a řídicích úloh*. Ostrava: 2003. 48s.

Diplomová práce VŠB-TU Ostrava.

LIPUS, M. *Programová podpora komunikace s mikropočítačovou jednotkou CTRL v prostředí MS Windows 95/98/NT*. Ostrava: 1999. 63s. Bakalářská práce VŠB-TU Ostrava.

LACKO, L. *ASP.NET pre začiatovníkov*, [online]. PDF formát. Dostupné z:

<<http://www.microsoft.com/cze/msdn/csebooks/default.aspx>>

MAHUGH, D. *Custom UI Editor Tool*, [online]. HTML format. Dostupné z:

<<http://openxmldeveloper.org/articles/CustomUIeditor.aspx>>

MELOUN, M., MILITKÝ, J. *Statistické zpracování experimentálních dat*. Praha: Plus, 1994. 839s. ISBN 80-85297-56-6.

MICROSOFT. 2006 A. *Developer Overview of the User Interface for the 2007 Microsoft Office System*, [online]. DOC formát. Dostupné z:

<<http://www.microsoft.com/downloads/details.aspx?familyid=5AE8EA78-6BA9-4DE4-AABD-2616D010CAA7&displaylang=en>>

MICROSOFT. 2006 B. *Office New User Experience: UI Style Guide for Solutions and Add-Ins*, [online]. DOC formát. Dostupné z:

<<http://www.microsoft.com/downloads/details.aspx?familyid=19E3BF38-434B-4DDD-9592-3749F6647105&displaylang=en>>

MICROSOFT. 2005. *Webový vývoj v ASP.NET 2.0 pomocí bezplatných Express nástrojů pro úplné začátečníky*, [online]. PDF formát. Dostupné z:

<<http://www.microsoft.com/cze/msdn/csebooks/default.aspx>>

NEUWIRTH, M. 2006. *Visual Studio Tools for Office 2005 Second Edition (VSTO 2005 SE)*, [online]. Webová prezentace. ISV Technical Readiness, Microsoft s.r.o., 2006. Dostupné z:

<<http://www.microsoft.com/cze/msdn/webcasts/default.aspx>>

NEUWIRTH, M. 2008. *Novinky ve VSTO 2008*, [online]. Webová prezentace. ISV Technical Readiness, Microsoft s.r.o., 2008. Dostupné z:

<<http://www.microsoft.com/cze/msdn/webcasts/default.aspx>>

NOSKIEVIČ, P. *Modelování a identifikace systémů*. 1. vyd. Ostrava: Montanex, 1999. 276s. ISBN 80-7225-030-2.

PETROUTSOS, E. *Myslíme v jazyku Visual Basic .NET – 2. díl*. 1. vyd. Praha: Grada Publishing, 2003. 540s. ISBN 80-7226-785-X.

RICE, F., GETZ, K. 2006 A. *Customizing the 2007 Office Fluent Ribbon for Developers (Part 1 of 3)*, [online]. HTML format. MCW Technologies, LLC, Microsoft Corporation, 2006. Dostupné z: <<http://msdn.microsoft.com/en-us/aa338202.aspx>>

RICE, F., GETZ, K. 2006 B. *Customizing the 2007 Office Fluent Ribbon for Developers (Part 2 of 3)*, [online]. HTML format. MCW Technologies, LLC, Microsoft Corporation, 2006. Dostupné z: <<http://msdn.microsoft.com/en-us/aa338199.aspx>>

RICE, F., GETZ, K. 2008. *Customizing the 2007 Office Fluent Ribbon for Developers (Part 3 of 3)*, [online]. HTML format. MCW Technologies, LLC, Microsoft Corporation, 2008. Dostupné z: <<http://msdn.microsoft.com/en-us/library/aa722523.aspx>>

ROSENBLUM, M. 2007. *Customizing the Excel 2007 Ribbon using VBA*, [online]. HTML format. Dostupné z: <<http://www.xtremevbtaalk.com/showthread.php?t=265636>>

SMUTNÝ, L. *Řízení teplovzdušného obvodu pomocí PC a mikropočítačové jednotky CTRL*. Ostrava : VŠB-TU Ostrava, 1998. Návod k laboratorní úloze. 12 s.

VÍTEČKOVÁ, M., VÍTEČEK, A. *Základy automatické regulace*. 1. vyd. Ostrava: VŠB-TU Ostrava, 2006. 200s. ISBN 80-248-1068-9.

WALKENBACH, J. *Microsoft Office Excel 2007 Bible*. Indianapolis: Wiley Publishing, 2007. 914s. ISBN 978-0-470-04403-2.

WALKENBACH, J. *Excel 2007 VBA Programming For Dummies*. Indianapolis: Wiley Publishing, 2007. 401s. ISBN 978-0-470-04674-6.

WALTUCH, M. 2009. *.NET Application Help*, [online]. HTML formát. Dostupné z: <<http://www.helpware.net/mshelp2/demo2/h1xNET.htm>>

WIKIPEDIE. 2009. *Metoda nejmenších čtverců – Wikipedie, otevřená encyklopedie*, [online]. HTML formát. Dostupné z: <http://cs.wikipedia.org/wiki/Metoda_nejmen%C5%A1%C3%ADch_%C4%8Dtverc%C5%AF>

Seznam obrázků

OBR. 1 PRACOVNÍ PROSTŘEDÍ PROGRAMU APROX SE ZADANÝMI DATY	12
OBR. 2 FORMÁT VÝSTUPU DAT Z PROGRAMU APROX	13
OBR. 3 ZOBRAZENÍ ZADANÝCH DAT (BODY) A JEJICH APROXIMACE SPOLU S KŘIVKOU ODCHYLEK	14
OBR. 4 PRACOVNÍ PROSTŘEDÍ PROGRAMU WINGAP S NAČTENÝMI DATY	16
OBR. 5 FORMÁT VÝSTUPU DAT Z PROGRAMU WINGAP	16
OBR. 6 ZOBRAZENÍ NAČTENÝCH DAT (ČERVENÁ) A JEJICH APROXIMACE (ZELENÁ)	18
OBR. 7 PRACOVNÍ PROSTŘEDÍ PROGRAMU JAP S VYGENEROVANÝMI DATY	19
OBR. 8 OKNO PRO PŘIPOJENÍ NA MYSQL DATABÁZI	20
OBR. 9 GRAF VYGENEROVANÉHO DEMA	22
OBR. 10 MICROSOFT .NET FRAMEWORK	25
OBR. 11 KOMPONENTY CLR	26
OBR. 12 DOPLŇKY APLIKACE MS EXCEL	27
OBR. 13 VÝBĚR PLATFORMY .NET FRAMEWORKU PŘI ZALOŽENÍ PROJEKTU	31
OBR. 14 INTEGROVANÉ PROSTŘEDÍ EXCELU PŘÍMO DO VISUAL STUDIA	33
OBR. 15 CUSTOM UI EDITOR	34
OBR. 16 ZAŘAZENÍ PRIMARY INTEROP ASSEMBLIES	34
OBR. 17 VSTO RUNTIME	35
OBR. 18 VZHLED A POPIS RIBBON PRVKU	38
OBR. 19 KÓD NUTNÝ PRO ZOBRAZENÍ RIBBON PRVKU	40
OBR. 20 RIBBON PRVEK A JEHO XML DOKUMENT	41
OBR. 21 VOLÁNÍ CALLBACK FUNKCÍ	42
OBR. 22 VLASTNOST STARTFROMSCRATCH NASTAVENA NA TRUE	43
OBR. 23 SOLUTION EXPLORER A „VYČIŠTĚNÍ“ PROJEKTU	44
OBR. 24 NOVÁ ZÁLOŽKA VISUAL APROX	47
OBR. 25 VZHLED SKUPINY <i>VSTUPNÍ DATA</i>	48
OBR. 26 PODPOROVANÉ FORMÁTY SOUBORU	48
OBR. 27 FORMULÁŘ PRO GENEROVÁNÍ DAT	49
OBR. 28 VZHLED SKUPINY <i>GRAFY</i>	50
OBR. 29 VZHLED SKUPINY <i>EXPERIMENTÁLNÍ IDENTIFIKACE</i>	50
OBR. 30 VZHLED SKUPINY <i>APROXIMACE</i>	51
OBR. 31 VZHLED SKUPINY <i>OSTATNÍ</i>	51
OBR. 32 FORMULÁŘ PRO ODSTRANĚNÍ ŘADY	52
OBR. 33 FORMULÁŘ PRO APROXIMACI ŘADY	52
OBR. 34 PRACOVNÍ PROSTŘEDÍ S DATY	53
OBR. 35 PRACOVNÍ PROSTŘEDÍ S VYKRESLENÝM GRAFEM	54
OBR. 36 PŘÍKLAD XML ZÁPISU NÁHODNĚ VYBRANÉHO SEŠITU APLIKACE MS EXCEL	55
OBR. 37 GRAF NAMĚŘENÝCH DAT (1. UKÁZKA)	56
OBR. 38 GRAF NAMĚŘENÝCH DAT (2. UKÁZKA)	57

Seznam tabulek

TAB. 1 PŘEHLED PODSTATNÝCH VÝHOD A NEVÝHOD ANALYZOVANÝCH PROGRAMOVÝCH MODULŮ	23
TAB. 2 VSTO 2008 POSKYTUJE TYTO MOŽNOSTI PRO OFFICE 2007.....	33
TAB. 3 MOŽNOSTI ZOBRAZENÍ STARŠÍHO ŘEŠENÍ V RIBBON PRVKU	38
TAB. 4 TABULKA MOŽNOSTÍ EDITACE RIBBON PRVKU V DESIGN MÓDU A POMOCÍ XML DOKUMENTU	39
TAB. 5 PROGRAMOVÉ PŘÍSTUPY PRO VYTVÁŘENÍ PROJEKTŮ.....	44
TAB. 6 OPTIMÁLNÍ UMÍSTĚNÍ POLOŽEK V RIBBON PRVKU.....	46

Přílohy:

Příloha 1: Ukázka XML dokumentu Ribbon prvku

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Definice XML schéma pro rozšíření Ribbon UI -->
<customUI xmlns="http://schemas.microsoft.com/office/2006/01/customui"
  onLoad="Ribbon_Load">
  <!-- Vlastnost Ribbon prvku pro zakázání ostatních záložek -->
  <ribbon startFromScratch="true">
    <!-- Kontextuální záložky -->
    <contextualTabs>
      <!-- Určení kontextuální záložky Nástroje grafu -->
      <tabSet idMso="TabSetChartTools">
        <!-- Vytvoření záložky VA - Aproximace, která bude
        umístěna před záložkou Návrh -->
        <tab id="VAAproximace" label="VA - Aproximace"
          insertBeforeMso="TabChartToolsDesign">
          <!-- Vytvoření skupiny Možnosti aproximace -->
          <group id="MoznostiAproximace" label="Možnosti aproximace">
            <!-- Vytvoření tlačítka Přímka, která má
            svou událost a větší celikost -->
            <button id="Primka" label="Přímka"
              size="large" onAction="Zprava" />
          </group>
        </tab>
      </tabSet>
    </contextualTabs>
    <!-- Záložky -->
    <tabs>
      <!-- Vytvoření záložky Visual Aprox a
      její umístění před záložku Domů -->
      <tab id="ZalozkaDoplanky" insertBeforeMso="TabHome"
        label="Visual Aprox">
        <!-- Vytvoření skupiny Vstupní data -->
        <group id="GrpVstupniData" label="Vstupní data">
          <!-- Vytvoření tlačítka Vytvořit data, které má ikonu
          z prostředí Excel, má svou událost a větší velikost -->
          <button id="BtnVytvoritData" imageMso="TableDrawTable"
            onAction="BtnVytvoritData_Click"
            label="Vytvořit data" size="large" />
          <!-- Vytvoření tlačítka Importovat data, které má ikonu
          z prostředí Excel, má svou událost a větší velikost -->
          <button id="BtnImportovatData" imageMso="ImportTextFile"
            onAction="BtnImportovatData_Click"
            label="Importovat data" size="large" />
        </group>
      </tab>
    </tabs>
  </ribbon>
</customUI>
```

Příloha 2: Ukázka kódu pro výpočet přechodové charakteristiky

```

Private Sub BtnPridatDataDoGrafu_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BtnPridatDataDoGrafu.Click
    _EA.ScreenUpdating = False
    _EA.Sheets("Data").Select()
    _EA.Range("A1").End(Excel.XlDirection.xlToRight).Offset(0, 1).Activate()
    Dim ay As Double
    Select Case TabControl.SelectedTab.Name
        Case "P1"
            _EA.ActiveCell.FormulaR1C1 = "h(t) P-1; K=" & TbxK1.Text & "; T=" & TbxT1.Text & "; "
            _EA.ActiveCell.Offset(1, 0).Activate()
            For i = 2 To NumericUDPocetBodu.Value + 2
                ay = CDBl(TbxK1.Text) * (1 - Math.Pow(Math.E, (-(1 / CDBl(TbxT1.Text)) * _EA.Cells.Range("A" & i).Value)))
                _EA.ActiveCell.Value = ay
                _EA.ActiveCell.Offset(1, 0).Activate()
            Next
        Case "P2"
            If CDBl(TbxKsi.Text) >= 0 And CDBl(TbxKsi.Text) < 1 Then
                _EA.ActiveCell.FormulaR1C1 = "h(t) P-2; K=" & TbxK1.Text & "; T=" & TbxT0.Text & "; ξ=" & TbxKsi.Text & "; "
                _EA.ActiveCell.Offset(1, 0).Activate()
                Dim C1, delta, omega, fi As Double
                delta = CDBl(TbxKsi.Text) / CDBl(TbxT0.Text)
                omega = (1 / CDBl(TbxT0.Text)) * Math.Sqrt(1 - Math.Pow(CDBl(TbxKsi.Text), 2))
                C1 = 1 / (omega * CDBl(TbxT0.Text))
                fi = Math.Atan(omega / delta)
                For i = 2 To NumericUDPocetBodu.Value + 2
                    ay = CDBl(TbxK1.Text) * (1 - C1 * Math.Pow(Math.E, (-(delta * _EA.Cells.Range("A" & i).Value)) * Math.Sin(a:=omega * _EA.Cells.Range("A" & i).Value + fi)))
                    _EA.ActiveCell.Value = ay
                    _EA.ActiveCell.Offset(1, 0).Activate()
                Next
            ElseIf CDBl(TbxKsi.Text) = 1 Then
                _EA.ActiveCell.FormulaR1C1 = "h(t) P-2; K=" & TbxK1.Text & "; T=" & TbxT0.Text & "; ξ=" & TbxKsi.Text & "; "
                _EA.ActiveCell.Offset(1, 0).Activate()
                For i = 2 To NumericUDPocetBodu.Value + 2
                    ay = CDBl(TbxK1.Text) * (1 - (1 + (1 / CDBl(TbxT0.Text)) * _EA.Cells.Range("A" & i).Value) * Math.Pow(Math.E, (-(1 / CDBl(TbxT0.Text)) * _EA.Cells.Range("A" & i).Value)))
                    _EA.ActiveCell.Value = ay
                    _EA.ActiveCell.Offset(1, 0).Activate()
                Next
            End If
        Case "I1"
            _EA.ActiveCell.FormulaR1C1 = "h(t) I-1; K=" & TbxK1.Text & "; T=" & TbxT1.Text & "; "
            _EA.ActiveCell.Offset(1, 0).Activate()
            For i = 2 To NumericUDPocetBodu.Value + 2
                ay = CDBl(TbxK1.Text) * (CDBl(TbxT1.Text) * Math.Pow(Math.E, (-(1 / CDBl(TbxT1.Text)) * _EA.Cells.Range("A" & i).Value) - 1) + _EA.Cells.Range("A" & i).Value)
                _EA.ActiveCell.Value = ay
                _EA.ActiveCell.Offset(1, 0).Activate()
            Next
    End Select
    _EA.ActiveCell.Offset(-1, 0).Select()
    _EA.ActiveCell.End(Excel.XlDirection.xlUp).Select()
    _EA.Columns.AutoFit()
    BtnNovaData.Enabled = True
    _EA.Sheets("Graf").Select()
    _EA.ScreenUpdating = True
End Sub

```

Příloha 1: Ukázka odvození výpočtu koeficientů polynomu 2. stupně

Při aproximaci polynomem $P_k = p_0 + p_1x + p_2x^2 + \dots + p_kx^k$ lze pro jeho koeficienty sestavit následující soustavu lineárních rovnic.

$$\begin{bmatrix} \sum_{i=1}^n x^{2k} & \cdot & \cdot & \cdot & \sum_{i=1}^n x^{k+1} & \sum_{i=1}^n x^k \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \sum_{i=1}^n x^{k+1} & \cdot & \cdot & \cdot & \sum_{i=1}^n x^2 & \sum_{i=1}^n x \\ \sum_{i=1}^n x^k & \cdot & \cdot & \cdot & \sum_{i=1}^n x & n \end{bmatrix} \cdot \begin{bmatrix} p_k \\ \cdot \\ \cdot \\ \cdot \\ p_1 \\ p_0 \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^n yx^k \\ \cdot \\ \cdot \\ \cdot \\ \sum_{i=1}^n yx \\ \sum_{i=1}^n y \end{bmatrix}$$

Tato soustava lineárních rovnic je tedy ve tvaru $Ax = b$ (kde A je matice, která je známa, b je vektor, také známý a x je vektor neznámý). V tomto případě nastává situace, kdy matice A je čtvercová, vektor b má stejný rozměr jako vektor x a tím pádem je k dispozici stejný počet rovnic jako neznámých. Vektor x lze pak vypočítat například pomocí inverzní matice $x = A^{-1}b$.

Postup pro polynom 2. stupně:

$$P_k = p_0 + p_1x + p_2x^2$$

$$A = \begin{bmatrix} \sum_{i=1}^n x^4 & \sum_{i=1}^n x^3 & \sum_{i=1}^n x^2 \\ \sum_{i=1}^n x^3 & \sum_{i=1}^n x^2 & \sum_{i=1}^n x \\ \sum_{i=1}^n x^2 & \sum_{i=1}^n x & n \end{bmatrix}; x = \begin{bmatrix} p_2 \\ p_1 \\ p_0 \end{bmatrix}; b = \begin{bmatrix} \sum_{i=1}^n yx^2 \\ \sum_{i=1}^n yx \\ \sum_{i=1}^n y \end{bmatrix}$$

$$A^T = \begin{bmatrix} \sum_{i=1}^n x^4 & \sum_{i=1}^n x^3 & \sum_{i=1}^n x^2 \\ \sum_{i=1}^n x^3 & \sum_{i=1}^n x^2 & \sum_{i=1}^n x \\ \sum_{i=1}^n x^2 & \sum_{i=1}^n x & n \end{bmatrix}$$

$$A^{-1} = \begin{bmatrix} \sum_{i=1}^n x^2 \cdot n - \sum_{i=1}^n x \cdot \sum_{i=1}^n x & -\left(\sum_{i=1}^n x^3 \cdot n - \sum_{i=1}^n x \cdot \sum_{i=1}^n x^2\right) & \sum_{i=1}^n x^3 \cdot \sum_{i=1}^n x - \sum_{i=1}^n x^2 \cdot \sum_{i=1}^n x^2 \\ -\left(\sum_{i=1}^n x^3 \cdot n - \sum_{i=1}^n x^2 \cdot \sum_{i=1}^n x\right) & \sum_{i=1}^n x^4 \cdot n - \sum_{i=1}^n x^2 \cdot \sum_{i=1}^n x^2 & -\left(\sum_{i=1}^n x^4 \cdot \sum_{i=1}^n x - \sum_{i=1}^n x^3 \cdot \sum_{i=1}^n x^2\right) \\ \sum_{i=1}^n x^3 \cdot \sum_{i=1}^n x - \sum_{i=1}^n x^2 \cdot \sum_{i=1}^n x^2 & -\left(\sum_{i=1}^n x^4 \cdot \sum_{i=1}^n x - \sum_{i=1}^n x^2 \cdot \sum_{i=1}^n x^3\right) & \sum_{i=1}^n x^4 \cdot \sum_{i=1}^n x^2 - \sum_{i=1}^n x^3 \cdot \sum_{i=1}^n x^3 \end{bmatrix}$$

$$x = A^{-1}b$$

$$\begin{bmatrix} p_2 \\ p_1 \\ p_0 \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^n x^2 \cdot n - \sum_{i=1}^n x \cdot \sum_{i=1}^n x & -\left(\sum_{i=1}^n x^3 \cdot n - \sum_{i=1}^n x \cdot \sum_{i=1}^n x^2\right) & \sum_{i=1}^n x^3 \cdot \sum_{i=1}^n x - \sum_{i=1}^n x^2 \cdot \sum_{i=1}^n x^2 \\ -\left(\sum_{i=1}^n x^3 \cdot n - \sum_{i=1}^n x^2 \cdot \sum_{i=1}^n x\right) & \sum_{i=1}^n x^4 \cdot n - \sum_{i=1}^n x^2 \cdot \sum_{i=1}^n x^2 & -\left(\sum_{i=1}^n x^4 \cdot \sum_{i=1}^n x - \sum_{i=1}^n x^3 \cdot \sum_{i=1}^n x^2\right) \\ \sum_{i=1}^n x^3 \cdot \sum_{i=1}^n x - \sum_{i=1}^n x^2 \cdot \sum_{i=1}^n x^2 & -\left(\sum_{i=1}^n x^4 \cdot \sum_{i=1}^n x - \sum_{i=1}^n x^2 \cdot \sum_{i=1}^n x^3\right) & \sum_{i=1}^n x^4 \cdot \sum_{i=1}^n x^2 - \sum_{i=1}^n x^3 \cdot \sum_{i=1}^n x^3 \end{bmatrix} \cdot \begin{bmatrix} \sum_{i=1}^n yx^2 \\ \sum_{i=1}^n yx \\ \sum_{i=1}^n y \end{bmatrix}$$

Polynom 2. stupně: $P_k = p_0 + p_1x + p_2x^2$

Koeficienty polynomu 2. stupně:

$$p_0 = \left[\left(\sum_{i=1}^n x^3 \cdot \sum_{i=1}^n x - \sum_{i=1}^n x^2 \cdot \sum_{i=1}^n x^2 \right) - \left(\sum_{i=1}^n x^4 \cdot \sum_{i=1}^n x - \sum_{i=1}^n x^2 \cdot \sum_{i=1}^n x^3 \right) + \left(\sum_{i=1}^n x^4 \cdot \sum_{i=1}^n x^2 - \sum_{i=1}^n x^3 \cdot \sum_{i=1}^n x^3 \right) \right] \cdot \sum_{i=1}^n y$$

$$p_1 = \left[-\left(\sum_{i=1}^n x^3 \cdot n - \sum_{i=1}^n x^2 \cdot \sum_{i=1}^n x \right) + \left(\sum_{i=1}^n x^4 \cdot n - \sum_{i=1}^n x^2 \cdot \sum_{i=1}^n x^2 \right) - \left(\sum_{i=1}^n x^4 \cdot \sum_{i=1}^n x - \sum_{i=1}^n x^3 \cdot \sum_{i=1}^n x^2 \right) \right] \cdot \sum_{i=1}^n yx$$

$$p_2 = \left[\left(\sum_{i=1}^n x^2 \cdot n - \sum_{i=1}^n x \cdot \sum_{i=1}^n x \right) - \left(\sum_{i=1}^n x^3 \cdot n - \sum_{i=1}^n x \cdot \sum_{i=1}^n x^2 \right) + \left(\sum_{i=1}^n x^3 \cdot \sum_{i=1}^n x - \sum_{i=1}^n x^2 \cdot \sum_{i=1}^n x^2 \right) \right] \cdot \sum_{i=1}^n yx^2$$